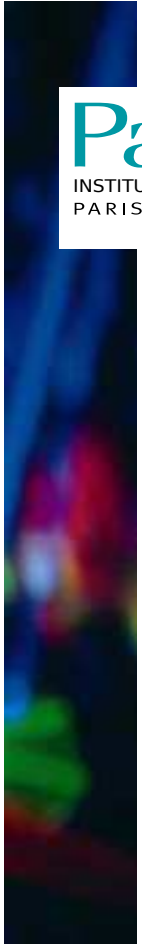
Modélisation de contextes pour l'annotation sémantique de vidéos

Nicolas Ballas



ParisTech
INSTITUT DES SCIENCES ET TECHNOLOGIES
PARIS INSTITUTE OF TECHNOLOGY

MINES ParisTech

École doctorale n°432: SMI - Sciences des Métiers de l'Ingénieur

Doctorat ParisTech

T H È S E

pour obtenir le grade de docteur délivré par

l'École nationale supérieure des mines de Paris
Spécialité Mathématiques et Systèmes

présentée et soutenue publiquement par

Nicolas Ballas

le 12 novembre 2013

Modélisation de contextes pour l'annotation sémantique de vidéos

Directeur de thèse: **Françoise Prêteux**

Co-encadrement de la thèse: **Bertrand Delezoide**

Jury

Mme Cordélia Schmid, Directeur de Recherche, INRIA	Président
M. Jean Ponce, Directeur de Recherche, ENS	Rapporteur
M. Georges Quénot, Directeur de recherche, CNRS	Rapporteur
M. Alexander Hauptmann, Senior System Scientist, CMU	Examinateur
M. Josef Sivic, Chercheur, INRIA	Examinateur
M. Marcin Dietiniecky, Chercheur, CNRS	Examinateur
Mme Françoise Prêteux, Directeur adjoint, Mines-ParisTech	Directeur
M. Bertrand Delezoide, Ingénieur de Recherche, CEA-LIST	Co-directeur

MINES ParisTech
Mathématiques et Sysèmes, CAOR - Centre de CAO et Robotique
MINES ParisTech, 60 Boulevard Saint-Michel 75006 Paris, France

THÈSE

Abstract

Recent years have witnessed an explosion of multimedia contents available. In 2010 the video sharing website YouTube announced that 35 hours of videos were uploaded on its site every minute, whereas in 2008 users were "only" uploading 12 hours of video per minute. Due to the growth of data volumes, human analysis of each video is no longer a solution; there is a need to develop automated video analysis systems.

This thesis proposes a solution to automatically annotate video content with a textual description. The thesis core novelty is the consideration of multiple contextual information to perform the annotation.

With the constant expansion of visual online collections, automatic video annotation has become a major problem in computer vision. It consists in detecting various objects (human, car...), dynamic actions (running, driving...) and scenes characteristics (indoor, outdoor...) in unconstrained videos. Progress in this domain would impact a wild range of applications including video search, video intelligent surveillance or human-computer interaction.

Although some improvements have been shown in concept annotation, it still remains an unsolved problem, notably because of the semantic gap. The semantic gap is defined as the lack of correspondences between video features and high-level human understanding. This gap is principally due to the concepts intra-variability caused by photometry change, objects deformation, objects motion, camera motion or viewpoint change...

To tackle the semantic gap, we enrich the description of a video with multiple contextual information. Context is defined as "the set of circumstances in which an event occurs". Video appearance, motion or space-time distribution can be considered as contextual clues associated to a concept. We state that one context is not informative enough to discriminate a concept in a video. However, by considering several contexts at the same time, we can address the semantic gap.

More precisely the thesis major contributions are the following:

- a novel framework that takes into consideration several contextual information: To benefit from mutiple contextual clues, we introduce a fusion scheme based on a generalize sparsity criteria. This fusion model automatically infers the set of relevent contexts for a given concept.

- a feature inter-dependences context modeling: Different features capture complementary information. For instance, Histogram of Gradient (HoG) focuses on the video appearance while the Histogram of Flow (HoF) collects motion information. Most of the existing works capture different feature statistics independently. By contrast, we leverage their covariance to refine our video signature.

- a concept-dependent modeling of space-time context: Discriminative information is not equally distributed in the video space-time domain. To identify the discriminative regions, we introduce a learning algorithm that determines the space-time shape associated to each individual concept.

- an attention context modeling: We enrich video signatures with biological-inspired attention maps. Such maps allow to capture space-time contextual information while preserving the video signature invariance to the translation, rotation and scaling transformations. Without this space-time invariance, different concept instances with various localizations in the space-time volume can result in divergent representations. This problem is severe for the dynamic actions which have dramatic space-time variability.

Resumé en Français

Les nouveaux comportements sociaux, transformations sociétales ainsi que la démocratisation des logiciels informatiques ont conduit à une explosion de la création de données. A cet effet, le domaine des “Big Data”, qui regroupe les méthodes pour capturer, traiter et analyser les données à large échelle, est devenu un sujet majeur des technologies de l’information au vu de ses implications économiques mais aussi étant donnée les questions de recherche sous-jacente à ce domaine [29, 122, 137]. Le contenu multimédia ne fait pas exception à la tendance “Big Data”. En effet, ces dernières années ont connu une explosion du contenu multimédia notamment avec le développement des caméras dans les téléphones mobiles. En 2013, plus de 100 heures de vidéo étaient rajoutées chaque seconde sur le site Youtube [51].

(a) Variabilité visuelle due á l’environement

(b) Variabilité visuelle due l’apparence des objets

Figure 0-1: Diversité du contenu visuel

Bien que le nombre de vidéos disponible à augmenter de manière drastique, les solutions pour leurs analyse automatique restent limitées. En effet, les systèmes de vision par ordinateur, qui ont pour but d’analyser et d’interpréter les données visuelles, sont loin d’égaler les capacités humaines [86]. Le principale difficulté de la vision par ordinateur est la forte variabilité du contenu visuel (cf. Figure 0-1) due à la fois à des

changement environnementaux (illumination, point de vue, occlusion, *etc.*) et à la forte diversité d'apparences des objets et/ou personnes. Bien que les humains arrivent à ignorer cette variabilité visuelle pour se concentrer sur les informations sémantiques contenues dans les donnés, les approches automatiques connaissent plus de difficultées

Dans cette thèse, on s'intéresse au problème d'analyse visuelle automatique à travers l'annotation d'action humaine dans les vidéos. Cette tâche, qui consiste à enrichir une vidéo avec une description textuelle exposant ses différentes actions, possède des implications pour deux nombreux domaines d'applications tels que le data mining, la vidéo-surveillance intelligente, ou les interactions homme-machine. Les approches s'attaquant à ce problème généralement représentent les vidéos avec des signatures bas-niveaux. De telles signatures résument les aspects clés d'une vidéo en capturant la distribution de ses motifs spatio-temporels. Ces signatures sont ensuite utilisées par des modèles de classification statistique qui infèrent la probabilité de présence d'une action. Bien que d'important progrès aient été réalisé ces dernières années, le problème d'annotation d'action reste non-résolu, à cause notamment de forte variabilité visuelle des contenus multimédia.

Cette thèse propose d'enrichir le modéle de classification statistique avec de multiples contextes. Nous définisons un contexte comme étant une description numérique d'une vidéo. Chaque signature bas-niveau capturant des informations sur une vidéo (apparences, mouvements, ou position spatio-temporelle) définit donc un contexte particulier. De plus, les contextes peuvent aussi être composés d'informations non-directement extraite des données multimédia de la vidéo, comme par exemple, des informations relatives à l'utilisateur ayant mis-en-ligne la vidéo, des information de géolocalisation [149]... Un contexte est donc un facteur qui caractérise un aspect particulier d'une vidéo. Notre hypothèse principale est qu'un seul contexte n'est pas assez discriminatif pour reconnaitre une action dans une vidéo. Néanmoins, en considérant conjointement plusieurs contextes, il est possible d'améliorer la reconnaissance d'action dans les vidéos.

En particulier, ce travail propose un modèle de classification prenant en considéra-

tion la complémentarité entre plusieurs contextes, ainsi que 3 nouveaux contextes de vidéo basés sur la covariance de caractéristiques locales, la modélisation de la forme spatio-temporelle des actions, et l'attention présente dans une vidéo

Classification basée contextes

Lors d'une première contribution, nous proposons un modèle de classification qui exploite la complémentarité entre plusieurs contextes. Notre modèle repose sur deux hypothèses: (i) l'utilisation de plusieurs contextes est nécessaire pour capturer la riche diversité d'un contenu multimédia (ii) certains contextes sont plus informative quand à la présence d'une action donnée dans une vidéo. Nous proposons donc un modèle qui détermine de manière automatique, quels sont les contextes importants pour une action.

Pour définir notre modèle, on se place dans le cadre de la classification supervisée binaire. On considère un jeu de donnée d'apprentissage $\mathbf{D} = \{\mathbf{X}, \mathbf{Y}\}$ composé N signautre basé contextes $\mathbf{X} = \{\mathbf{X}_1, ..., \mathbf{X}_N\}$, et le vecteur de labels $\mathbf{Y} \in \{0, 1\}^N$, indiquant la présence ou l'absence d'une action dans une vidéo.

Chaque signature basée contexte est la concaténation de C contexte différant, *i.e.* $\mathbf{X}_i = [\mathbf{X}_i^1, \ldots, \mathbf{X}_i^C] \in \mathbb{R}^{1\times D}$ ou $\mathbf{X}_i^c \in \mathbb{R}^{1\times D_c}$ est le c-th contexte de la i-th vidéo. On cherche a apprendre un modèle linéaire, définit par le vecteur $\mathbf{W} \in \mathbb{R}^{1\times D}$ et le terme de bias $b \in \mathbb{R}$. Notre modèle linéaire $\mathbf{W}$ peut être décomposé en plusieurs sous-groupe de coéfficiants $\mathbf{W} = [\mathbf{W}_1, ..., \mathbf{W}_C]$, où $\mathbf{W}_c \in \mathbb{R}^{1\times D_c}$ sont les coefficients du modèle corrélé au c-th contexte de notre représentation. Notre modèle $(\mathbf{W}, b)$ capture l'apparence d'une action en minimisant la fonction objectif suivante:

$$\arg\min_{\mathbf{W},\mathbf{b}} O(\mathbf{W}, b, \mathbf{D}) = \sum_{i=1}^{N} L(Y_i, \sum_{c=1}^{C} \mathbf{W_c} \mathbf{X_i^c} + b) + \lambda\Omega(\mathbf{W}). \tag{1}$$

. Dans (1), L est une fonction de perte qui pénalise les prédictions incorrectes du modèle $(\mathbf{W}, b)$ et Ω un terme de régularisation qui contraint la complexité de notre modèle pour éviter le sur-apprentissage.

Pour prendre en compte la structure multiple contextes de notre représentation

de vidéo, on induit des contraintes d'éparsité de groupe dans notre terme de régularisation:

$$\Omega(\mathbf{W}) = \|\mathbf{W}\|_{2,p}^{p} = \sum_{c=1}^{C} \|\mathbf{W}_c\|_2^{\frac{1}{p}} \tag{2}$$

.

Une norme $\|.\|_{2,p}$ est une combinaison d'une norme $\|.\|_p$ entre les différents contextes et d'une norme $\|.\|_2$ à l'intérieur de chaque contexte. Cela permet à notre modèle de selectionner un nombre limité de contexte à travers la norme $\|.\|_p$, tout en exploitant les corrélations implicites entre les différents éléments composant un contexte grâce à la norme $\|.\|_2$. Le paramètre p nous permet de contrôler l'éparsité (nombre de contextes sélectionnés) de notre modèle.

Contexte de Covariance

Dans une deuxième contribution, nous proposons un contexte qui caractérise la covariance des caractéristiques spatio-temporelle locales [197] dans les vidéos. La plupart des signatures [102, 196] décrivent une vidéo à travers plusieurs descripteurs locaux. Ces descripteurs capturent différentes informations complémentaires (apparence, mouvement, accélération). Néanmoins, les signatures de l'état de l'art se focalisent généralement sur les statistiques de premier ordre de ces descripteurs. Ils ne prennent pas en compte leurs inter-relations.

Les inter-relations entre différent descripteurs locaux, caractérisant conjointement différentes modalités d'informations, pourraient permettre une meilleure description des vidéos. Pour évaluer l'impacte de ces inter-relations, cette thèse introduit des contextes de covariance capturant les statistiques du second ordre (moyenne et maximum) des descripteurs. De plus, les statistiques de second ordre ayant une forme matricielle, un modèle bi-linéaire, tirant profit de la structure 2D des contextes, est utilisé pour effectuer la classification. Ce modèle bi-linéaire est intégré à notre classification multi-contextes présenté précédemment (cf Figure 0-2).

Nous validons l'utilité de ce contexte par l'expérimentation sur les jeux de données

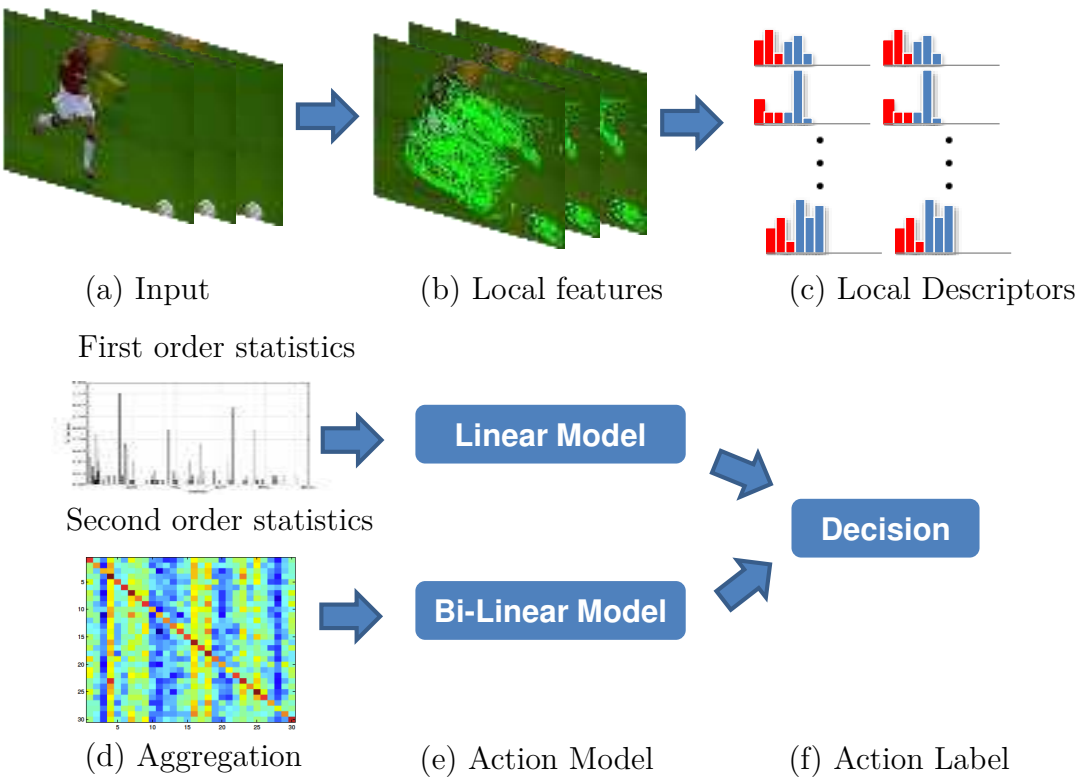


Figure 0-2: Synopsis of the covariance and BoW combinaison.

	BoW	cov-avg	cov-max	BoW + cov-avg	BoW + cov-max
KTH	93.7	94.6	**95.4**	94.9	94.1
HMDB	41.6	44.5	45.0	**51.1**	49.0

Table 1: Precision moyenne des contextes BoW et covariances (ainsi que leur combinaison) sur KTH et HMDB.

KTH et HMDB [95, 162]. Les trajectoires denses sont utilisées en tant que descripteur locaux [197]. Leur statistiques sont capturées en utilisant la représentation type sac-de-mots (BoW) [170] qui tend à considérer les statistique du premier ordre (Baseline) et en utilisant les contextes de covariances. Les résultats (cf Table 1) montrent la pertinence de notre approche. En effet, la combinaison des représentations sac-de-mots et covariance obtient toujours des meilleurs performances comparativement à la représentation sac-de-mots considérée seul, avec un gain allant jusqu'a 22%. Néanmoins, l'utilisation des matrices de covariance entraine aussi une augmentation non-négligeable de la dimensionnalité de notre représentation.

Contexte Spatio-Temporel

Periodic 2x2 Segmentation Grid Learned Adaptive Grid

Figure 0-3: Illustration desc grilles de segmentation fixe et des grille de segmentation adaptative. Les gilles adaptive sont capable de suivre approximativement une action à travers le temps dans une vidéo.

Cette thèse propose aussi un contexte capturant la localisation des descripteurs locaux. En effet, il a été démontré que la localisation de descripteurs locaux apporte des informations discriminatives pour la classification d'une action [102]. L'état de l'art utilise des grilles de segmentation fixes [102, 106] pour capturer ces informations. Ces grilles fixes sont prédéfinit, elles ne prennent pas en compte la localisation usuelle de l'action dans la vidéo. En conséquence, elles peuvent ne pas être optimales pour capturer le contexte spatio-temporel d'une action donnée (cf. Figure 0-3). Pour répondre à ce problème, nous proposons d'apprendre les grilles de segmentation directement à partir des vidéos d'apprentissage. Il en résulte des grilles qui s'adaptent aux changements de localisation d'une action.

Context	Adaptive Accuracy	Gain comparé à Grilles Fixes
Action Statique	87.9	−4%
Action Dynamique	85.3	12.6%
Statique + Dynamique	86.3	5.7%

Table 2: Analyse des résultat sur le jeux de données Youtube.

Une évaluation empirique sur 4 jeux de données montre que notre approche est plus performante que les grilles fixes avec un gain moyen 6%. Pour comprendre l'apport

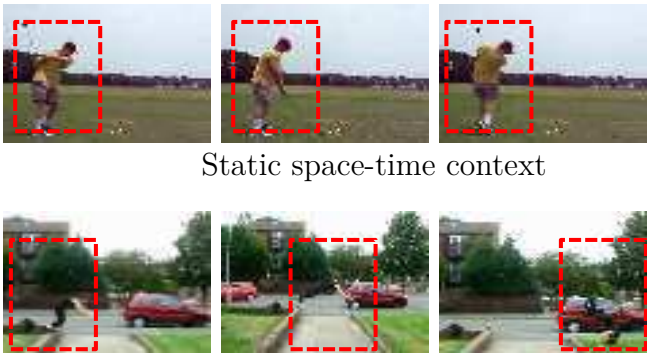


Figure 0-4: Action Statique vs Action Dynamique.

de notre approche, on classe les actions en deux catégories, les actions statiques et les actions dynamiques. Les actions statiques ont une localisation stable dans le temps alors que les actions dynamiques voient leurs positions variées au cours de la vidéo comme l'illustre la figure 0-4. En analysant les résultats sur le jeu de donnée UCF-Youtube [113], on observe que les grilles adaptatives sont particulièrement efficace pour modéliser le contexte spatio-temporel associé aux actions dynamiques. Cela tend à montrer que notre approche est capable de suivre approximativement une action dans le temps. Pour les actions statiques, les grilles adaptatives n'apportent pas d'information complémentaire par rapport aux grilles fixes.

Contexte d'attention

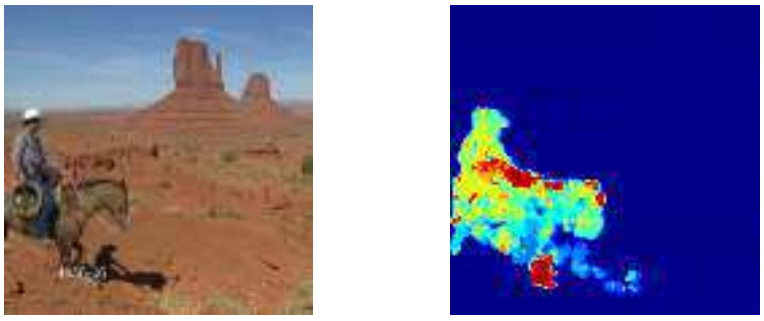


Figure 0-5: Exemple de carte de saillance basée sur le mouvement d'une trame vidéo.

Dans une quatrième contribution, nous proposons de tirer profit des informations d'attention dans une vidéo. L'attention permet de mettre en valeur les parties à priori discriminante dans un contenu visuel [67]. Étant donné une image, l'attention produit une carte de saillance identifiant les régions qui attire le regard humain (cf. Figure 0-5).

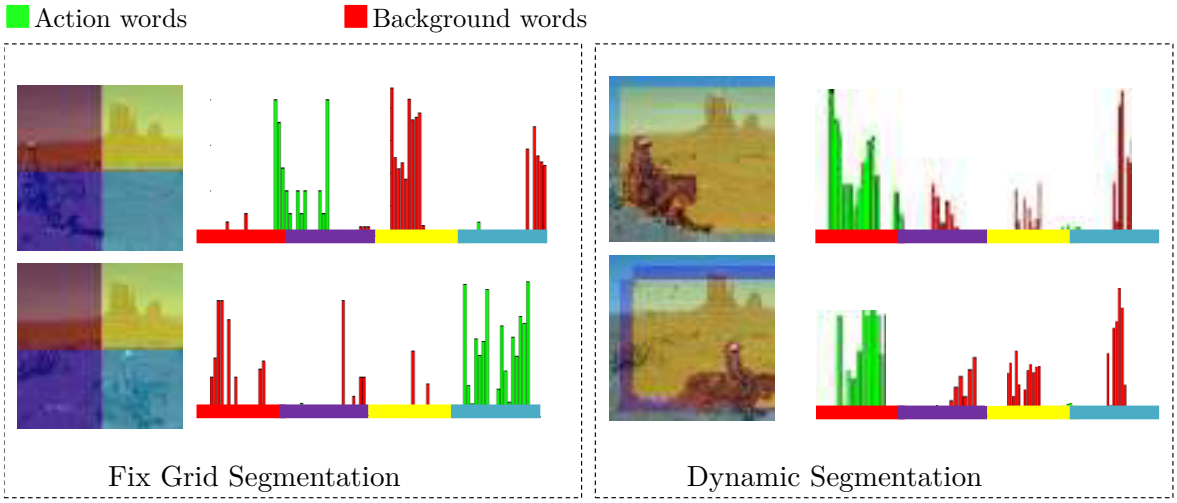


Figure 0-6: Segmentation avec une grille fixe vs Segmentation basée sur la saillance. A cause du mouvement de l'action dans le temps, une grille divisant l'espace en 2x2 cellules va melanger les information associer au fond et à l'action dans ses différentes cellules. Différemment, la segmentation dynamique suit l'action au cours du temps, en effet l'action reste la zone visuelle prédominante tout au long de la vidéo.

Pour bénéficier de l'attention, cette thèse introduit un contexte qui capture la distribution de caractéristiques locales, non pas dans des sous-domaines géométrique, mais dans des sous-espaces de saillance. Ce contexte permet de caractériser différemment les régions saillante et non-saillante dans une vidéo. En d'autre terme, plutôt une grille spatial prédéfini [102], ce contexte propose de segmenter les vidéos en utilisant les informations de saillance pour ainsi obtenir une segmentation spécifique par vidéo (cf Figure 0-6). De plus, les fonctions de saillance étant invariantes aux transformations spatio-temporelles (translation, rotation...), le contexte d'attention préserve cette robustesse. Cela est primordial pour l'annotation d'actions. Les actions humaines sont en effet à des variations de positions assez importante dans le temps due à leurs dynamiques.

	KTH	UCF50 5 folds	UCF50 25 folds	HMDB
BoW	93.7	86.7	85.3	41.6
Grilles Fixes	94.0	91.2	89.3	44.0
Spa 3x3x3	93.8	91.4	89.1	44.1
Attention	94.4	92.5	91.3	48.5
Attention + Grille Fixe	**94.6**	**94.1**	**92.8**	**51.8**

Table 3: Précision moyenne des contexts sac-de-mots (BoW), Grilles Fixes et Attention sur plusieurs HMDB, UCF50 et KTH..

Une évaluation empirique montre l'apport de ce contexte. Nous comparons notre approche avec les représentations sac-de-mots [170] et le grilles de segmentation fixes [102]. Les résultats (Table 3) montrent que contexte d'attention obtient les meilleur performances parmi ces représentation. De plus, la combinaison des grilles fixes et du contexte d'attention, utilisant le modèle de classification multi-contextes, ajoute un gain de performance (7% sur HMDB). Cela montre que la segmentation spatiale et la segmentation dans le domaine de saillance sont complémentaires.

Conclusion

La définition d'une représentation intermédiaire, ou contexte, est primordial pour l'annotation automatique d'action dans les vidéos. Une telle représentation doit mettre en valeur les informations discriminantes associées aux actions tout en étant robuste à la variabilité non-informative, inhérente aux contenus multimédia. Dans cette dissertation, nous avons proposé trois nouveaux contextes de vidéo (covariance, spatio-temporel et attention) qui nous ont permis de montrer que:

- les statistiques d'ordre supérieur des caractéristiques locales améliore le pouvoir discriminatif de la représentation;
- le contexte de localisation des caractéristiques locales dépend des actions;
- preserver l'invariance aux transformations spatial (translation, rotation. . .) tout en prenant en considération les informations de localisation spatio-temporel permet d'améliorer les performances de classification, comme l'a montré le contexte

	State-of-art		Thesis		Gain
UCF-101 [178]	44.5	[195] 2013	mutiple contexts	**87.7**	92%
HMDB [95]	57.1	[198] 2013	multiple contexts	**53.3**	-
UCF-50 [157]	84.5	[198] 2013	attention contexts	**92.7**	9%
UCF-Youtube [113]	84.0	[197] 2011	space-time context	**86.3**	4%
KTH [162]	94.5	[49] 2011	covariance context	**95.5**	1%
UT-Interaction 1 [160]	84.0	[144] 2012	space-time context	**91.3**	9%
UT-Interaction 2 [160]	86.0	[144] 2012	space-time context	**95.0**	11%

Table 4: Principaux résultat de la thèse (Précision Moyenne).

d'attention.

Cette thèse a aussi obtenu des résultat compétitif sur plusieurs jeux de données, comme le montre la Table 4. Ces résultats ont été obtenus en combinant différents contextes avec notre modèle de classification incluant des contraintes d'éparsité de groupes. Cela tend à vérifier que l'utilisation de plusieurs contextes est nécessaire pour capturer la diversité d'un contenu multimédia. De plus le gain du à l'éparsité montre certains contextes sont plus informative quand à la présence d'une action donnée dans une vidéo. La représentation optimale d'une vidéo dépend donc de son contenu.

Acknowledgments

I am indebted to many people who both directly and indirectly contributed to this dissertation. First of all, I thank my advisors, Françoise Prêteux and Bertrand Delezoide, for their unwavering guidance and support during my years as their student. Françoise kindness, optimism and expertise pushed me to take full advantage of my PhD experience. I am also very grateful to Bertrand which openness, enthusiasm, and scientific knowledge, as well as his cinematographic expertise, definitly had a strong impact on my thesis research work, but also on my mindset. Many thanks Alexander Hauptmann his kind welcome and advice during my stay in the CMU Informedia team. It was an outstanding experience, both academic and personal perspectives.

I am also grateful to my thesis commitee members, Cordélia Schmid, Jean Ponce, Georges Quénot, Alexander Hauptmann, Josef Sivic, Marcin Dietiniecky, for their interests and insightful comments about my work.

Finally, I am obliged to all the people I had the chance to collaborate with during those last few years: Adrian Pospescu, Adrien Chan-hoh-tong, Amel Znaidia, Dhouha Bouamor, Hamid Fard, Hervé Le-Borgne, Jim Braux-Zin, Lan Zhen-zhong, Ludovic Jean-louis, Wei Wang, Yang Cai and Yi Yang. All the novel ideas in thesis originate in discussions and exchanges with those bright persons.

"on a beau dire ce que qu'on voit, ce qu'on voit ne loge jamais dans ce qu'on dit"

Les mots et les choses, Michel Foucault

Contents

Abstract 3

Resumé 5

Acknowledgments 15

Contents 17

List of Figures 23

List of Tables 29

1 Introduction 33

1.1 Automated Multimedia Annotation 35

1.1.1 Concept Type 35

1.1.2 Video Type 37

1.2 Thesis Main Contributions 39

2 Related Work and Experimental Datasets 43

2.1 Visual Data Representation 44

2.1.1 Holistic Representation 45

2.1.2 Local Representation 46

2.1.3 Pose-Estimation based Representation 54

2.1.4 Semantic Representations 56

2.2 Concept Modeling 57

2.2.1 Linear and Kernel Methods . . . 57
2.2.2 Graphical model . . . 60
2.2.3 Information Fusion . . . 61
2.3 Experimental Datasets . . . 63
2.3.1 UT-interaction Datasets . . . 63
2.3.2 KTH Dataset . . . 65
2.3.3 UCF-Youtube, UCF-50, UCF-101 Datasets . . . 66
2.3.4 HMDB Dataset . . . 68
2.4 Conclusion . . . 70
2.4.1 Video Representations . . . 70
2.4.2 Concept Modeling . . . 71

3 Contribution: A Contextual View of Video Annotation 73
3.1 Motivation . . . 73
3.2 Context . . . 76
3.3 Developed Framework . . . 79
3.3.1 Model . . . 79
3.3.2 Problem Formulation . . . 79
3.3.3 Energy Based Modeling . . . 80
3.3.4 From Multiple Contexts to Concept: Generalized Sparsity Regularization . . . 82
3.3.5 Optimization . . . 84
3.3.6 Proof of Error Convergence . . . 85
3.4 How to apply the framework: WSVM instantiation . . . 87
3.4.1 Model . . . 87
3.4.2 A First Application . . . 88
3.5 Conclusion . . . 91

4 Feature Covariance Context 93
4.1 Motivation: Improving the Representation Discriminative Capability 94
4.1.1 Action Representation . . . 94

4.1.2 Bag-of-Words: First-Order Statistics . . . 95
4.1.3 Covariance: Higher-Order Statistics . . . 97
4.1.4 Our Contributions: . . . 98
4.2 Related Work . . . 99
4.2.1 Covariance Representation . . . 100
4.2.2 Bi-linear model . . . 101
4.3 Covariance Context . . . 102
4.4 Covariance Model . . . 105
4.4.1 Limitation of Linear Model for Covariance Matrices . . . 105
4.4.2 Multi-Compound Bi-Linear Model . . . 107
4.4.3 Integration in the Multiple Context Cues Model . . . 111
4.5 Covariance Context Added Value: Evaluation . . . 111
4.5.1 Implementation Detail . . . 111
4.5.2 Does Covariance Information bring discriminative information? 112
4.5.3 Analysis on a Constrained Dataset . . . 113
4.5.4 Analysis on Real World Video Datasets . . . 114
4.5.5 Is Covariance Matrix Structure Relevant for Classification? . . 118
4.5.6 Are Covariance and First-Order Representation Complementary? 120
4.6 Conclusion . . . 123

5 Task-Specific Space-Time Context 125

5.1 Motivation: Task-Specific Segmentation . . . 125
5.1.1 Local Features Space-Time Context . . . 126
5.1.2 Space-Time Context in BoW Representation . . . 127
5.1.3 Our Contribution . . . 128
5.2 Related Work . . . 129
5.3 Leveraging Viewpoint Repeatability to Learn Task-Specific Segmentation Grid . . . 131
5.4 Action Specific Pooling . . . 134
5.4.1 Generic Intermediate Representation . . . 135

5.4.2 Action-Specific Intermediate Representation 136
5.5 Identifying Informative Regions in Videos 136
5.5.1 Region Extraction 137
5.5.2 Position Extraction 138
5.6 Task-Specific Space-Time Context Modeling 139
5.6.1 Leveraging Structural Information 140
5.6.2 Leveraging Appearance Information 141
5.6.3 Optimization 142
5.7 Relevance of Adaptive Grids: Evaluation 143
5.7.1 Experimental Setting 143
5.7.2 Does the Space-Time Context Relevant Help in Improving Action Annotation? 144
5.7.3 Adaptive Grids: Proof of Concept 147
5.7.4 Adaptive Grids: Unconstrained Data 149
5.8 Conclusion 155

6 Biological-Inspired Attention Context **157**
6.1 Motivation: Retaining the Space-Time Invariance 157
6.1.1 Visual Attention Context 158
6.1.2 Attention Context and Space-Time Information 159
6.1.3 Our Contribution 161
6.2 Related Work 161
6.3 Space-Time Robust Representation 163
6.3.1 Content Driven Pooling 164
6.3.2 Saliency Measures 167
6.3.3 Space-Time Invariance Property 170
6.4 Top-Down Weighting 172
6.5 Attention Context Performances: Evaluation 174
6.5.1 Experimental Setting 175
6.5.2 When Do Saliency Cues Help for Action Recognition? 175

6.5.3 Are the Saliencies Complementary? . . . 180
6.5.4 Parameters Evaluation . . . 181
6.6 Conclusion . . . 182

7 Evaluation of Multiple Contexts Representation 185
7.1 Experimental Setting . . . 185
7.2 Individual Context Evaluation . . . 187
7.3 Context Combination Evaluation . . . 189
7.3.1 Combination Model . . . 189
7.3.2 Technical Details . . . 189
7.3.3 Combination Results . . . 190
7.3.4 Sparsity Impact . . . 192
7.4 Comparison with State-of-art . . . 193
7.5 Conclusion . . . 194

8 Conclusion 195
8.1 Key Contributions and Immediate Perspectives . . . 195
8.1.1 Covariance Context . . . 196
8.1.2 Task-Specific Space-Time Context . . . 197
8.1.3 Attention Context . . . 198
8.1.4 Multiple-Contexts Classification Framework . . . 199
8.2 Future Directions . . . 200

A Video Segmentation 203
A.1 Gibbs Point Process Model For Segmentation . . . 204
A.2 Optimization . . . 205

Bibliography 209

List of Figures

0-1 Diversité du contenu visuel 5

0-2 Synopsis of the covariance and BoW combinaison. 9

0-3 Illustration desc grilles de segmentation fixe et des grille de segmentation adaptative. Les gilles adaptive sont capable de suivre approximativement une action à travers le temps dans une vidéo. 10

0-4 Action Statique vs Action Dynamique. 11

0-5 Exemple de carte de saillance basée sur le mouvement d'une trame vidéo. 11

0-6 Segmentation avec une grille fixe vs Segmentation basée sur la saillance. A cause du mouvement de l'action dans le temps, une grille divisant l'espace en 2x2 cellules va melanger les information associer au fond et à l'action dans ses différentes cellules. Différemment, la segmentation dynamique suit l'action au cours du temps, en effet l'action reste la zone visuelle prédominante tout au long de la vidéo. 12

1-1 Double variability of computer vision 34

1-2 Video Annotation System. 35

1-3 Concept Taxonomy. 36

1-4 Video Taxonomy. 38

2-1 Taxonomy of image and video representations. 44

2-2 Examples of silouette based holistic representation. 45

2-3 Local Representation Synopsis. 47

2-4 Static Features Sampling Strategies (coutesy of [186]). 48

2-5 Space-Time Interest Point. Green zone correspond to the human silhouette, black zones are the detected salient regions (courtesy of [100]). 49

2-6 Trajectory features. 51

2-7 Synopsis of the Bag-of-Word model [170]. Local features are extracted from images, then quantized into a visual codebook. An image is then represented as a distribution of codebook words. 53

2-8 Exemple of 3D segmentation grid (courtesy of Laptev [100].) 54

2-9 Pose Estimation for Action Recognition (courtesy of Yang [211]). . . . 55

2-10 Two layers undirected graphical model (courtesy of Hauptmann [60]). 61

2-11 Frame samples from the UT-interaction datasets. 65

2-12 Frame samples from the KTH dataset. 66

2-13 Frame samples from the UCF datasets. 67

2-14 Frame samples from the HMDB dataset. 68

2-15 Distribution of the various conditions for the HMDB videos (courtesy of Kuehne [95]). a) visible body part, b) camera motion, c) camera view point, and d) clip quality. 69

3-1 Example of *car* image under different viewpoint and illumination parameters. 74

3-2 Illustration of a Bag-of-Words context. 77

3-3 Framework Synopsis. 80

3-4 Per class average accuracy on the HMDB datasets. 90

4-1 Decomposition of an action into spatio-temporal regions. Red rectangles identify the main action regions while red arrows correspond to the region principal motion. 95

4-2 Illustration of the covariance discriminative capacity. We consider an action recognitions problem with three classes. We extract the Histogram of Gradient and Histogram of Flow of video local trajectory features. We aggregate the local descriptors per video using simple first order statistic (average) and covariance. We apply a Linear Discriminant Analysis on both aggregation methods. This figure shows that the separation between the different classes is more apparent within the covariance representation. 98

4-3 Synopsis of the covariance and BoW combination. 99

4-4 Mean, max and average covariance pooling applied to two synthetic sets of local descriptors. Descriptors are distributed accordingly to multi-dimensional Gaussian having the same mean but different covariances. While mean and max pooling don't exhibit strong differences between the two distributions, covariance pooling is able to capture the distribution specificities. 104

4-5 Illustration of the KTH Running and Jogging actions. 113

4-6 Illustration of few HMDB actions. 114

4-7 Feature clutter illustration on the HMDB dataset. 114

4-8 Per class average accuracy on the HMDB datasets. 116

4-9 Test and training accuracies for different compound numbers for the bi-linear SVM applied on the the KTH and HMDB dataset. 119

4-10 KTH dataset. 121

4-11 HMDB dataset. 122

5-1 Surf and Jetski Frame Examples. 126

5-2 Spatial Pooling: the video volume is divided in different space-time cell according to a regular grid, and, one BoW representation is computed inside each cell. 127

5-3 Static vs Dynamic space-time context. Static space-time context remains stable over time while dynamic context knows variation. 128

5-4 Illustration des fixed versus task-specific grids. Task-specific grid coarsely follows the action through time. 129

5-5 Illustration of viewpoints repeatability. Red rectangles indicate the video discriminative regions. 132

5-6 Synopsis of action-specific-based recognition: green blocks correspond to our contributions. 133

5-7 Illustration of action-specific pooling. 134

5-8 Synopsis of motion region positions extraction. 137

5-9 Cluster Position Extraction. 139

5-10 Average segmentation grids of the segmentation regions of the "Shoot, "Swing" and "Juggle" action on the YouTube dataset. A 20x20x10 regular grid is used to quantize the segmented region positions. . . . 140

5-11 Evaluation of fixed regular grids on HMDB. 146

5-12 Adaptive Grid learned for *Shake-Hand*. First line: frame examples of a *Shake-Hand* action sampled at different time in a video. Second line: 4x4x4 Adative Grid learned for the *Shake-Hand* action. Only grid the cuboids with a reponse strenght superior to 0.1 are displayed. 148

5-13 4x4x4 Adaptive Grids on UT-interaction 2 dataset. Each line correspond to one UT-interaction 2 action. In each column, the first image shows a video example sampled at different time, the second image displays the heat map of the action Adaptive Grid. 149

5-14 Youtube actions with static space-time context. 150

5-15 Youtube actions with dynamic space-time context. 150

5-16 Per action Average Accuracy. 154

5-17 Parameters impact on HMDB. 154

6-1 Example of motion saliency map estimating the visual attention of an input video frame. 158

6-2 Space-time context importance: "Soccer" and "Running" are likely to be distinguished by the area surrounding the human legs in the video lower part while "Clap" and "Wave" are more easily distinguished by the upper-bodies. 159

6-3 Space-time variance: actions can be subject to localization variance due to camera viewpoint change in different videos. Even within a single video sequence, the action area is prone to change among frames. 160

6-4 Space-time robustness importance. Due to the action shift, 2x2 grid results in spatial BoWs having a low similarity despite representing the same action. Pooling using dynamic segmentation remains robust to the action space-time variance while still modeling the feature space-time context. 164

6-5 Space-Time Invariant Pooling. By segmenting in the saliency space, accordingly to their sailent rank, our representation remains invariant to global space-time transforms. 165

6-6 Prominent areas highlighted by the different saliency measures. Red contour indicates which saliency function obtain the best overlap with the actual action localization. 167

6-7 Different *structural primitives* highlights difference space-time region in the video. Using top-down information, we want to select the region that fit the action. 172

6-8 Attention Context Combination. 173

6-9 Impact of the vocabulary dimension. Average accuracy is reported. . 176

6-10 Prominent $\mathbf{W}_g$ groups in $\mathbf{W}$. The left column contains the reference frames. The middle column shows the extracted trajectories. The right column represents only the trajectories associated to the action most relevant *structural primitive*, i.e., the trajectories associated with the group $\mathbf{W}_g$ having the highest $\|.\|_2$ norm in $\mathbf{W}$. The most relevant *structural primitive* can be computed using cornerness, motion or light saliency depending on the action. 177

6-11 Per action average accuracy on HMDB. 178
6-12 Action Properties (courtesy of Kuehne [95]). 179
6-13 Impact of different parameters on the HMDB dateset. Average accuracy is reported. 181
6-14 Evaluation sparse feature weighting regularizer for the "Flic Flac" action on HMDB. On the left, $\|\mathbf{W}_g\|_2$ are displayed, for $p = 2$ or 1.5. On the right, features corresponding to two $\mathbf{W}_g$ groups are shown. 182

7-1 Evaluation of the sparsity parameter p on the HMDB dataset. 192

8-1 Deep Learning vs Engineered Representations. 200

A-1 Illustration of our video signatures computation. First local trajectory features are extracted from a video input (figure A-1b). Then trajectory cluster are computed through clustering (figure A-1c). Finally, we take advantage of the tunnel features spatio-temporal positions to obtain our final video segmentation (figure A-1d). 203
A-2 Some trajectory segmentation results. 206

List of Tables

1 Precision moyenne des contextes BoW et covariances (ainsi que leur combinaison) sur KTH et HMDB. 9

2 Analyse des résultat sur le jeux de données Youtube. 10

3 Précision moyenne des contexts sac-de-mots (BoW), Grilles Fixes et Attention sur plusieurs HMDB, UCF50 et KTH.. 13

4 Principaux résultat de la thèse (Précision Moyenne). 14

1.1 Overview of the thesis results on publicly available datasets. Average Accuracy is reported. 40

2.1 Holistics representation. 46

2.2 Static Local Features. 49

2.3 Short-Term Time Features. 50

2.4 Long-Term Time Features. 51

2.5 Pose Estimation based representation. 55

2.6 Pose Estimation based representation. 56

2.7 Datasets overview in term of Concepts number, Videos Number, Viewpoint Change, Camera motion and Background clutter. 64

2.8 Action Type for Human-Action Datasets. 64

2.9 Results on UT-interaction. 65

2.10 Results on KTH. 66

2.11 Results on UCF-Youtube. 67

2.12 Results on UCF50. 67

2.13 Results on UCF101. 68

2.14 Results on HMDB. 69
2.15 Synopsis of the video representations. 70

3.1 Taxonomy of existing methods in term of context categories. 78
3.2 Evaluation of the different spatial context HMDB dataset. 89
3.3 WSVM evaluation. 89

4.1 Summary of other approaches relying on feature covariance. SVM stands for Support Vector Machine, NN stands for Nearest Neighbors. 101
4.2 Average Accuracy for the BoW, cov-avg and cov-max representation. 113
4.3 Per action average accuracy on the KTH dataset. 113
4.4 Average accuracies of the intra and inter-descriptor covariances on the HMDB dataset for the max covariance pooling. It clearly highlights that the covariance inter-descriptors is more discriminative than the covariance intra-descriptor. Covariance is therefore especially performant when several descriptors are considered. 117
4.5 Spearman ρ factor of the different aggregation schemes. 118
4.6 Average accuracies of linear and bi-linear model for the cov-avg signatures on the KTH and HMDB dataset. 118
4.7 Comparaison of individual and combination performances. Average accuracies are reported. 121

5.1 Comparison of our approach with state-of-arts. 130
5.2 Average Accuracy on UT-Interaction 1 and 2. 148
5.3 Average Accuracies on the YouTube dataset. 151
5.4 Dynamic Spatial Context vs Static Spatial Context accuracy and performance gain on the YouTube dataset. 151

6.1 Comparison with other pooling methods taking into account the space-time context. 162
6.2 Comparison with other works using visual attention. 163

6.3 Average accuracies of BoW, structural-BoWs. Mo, Li and Co correspond respectively to Motion, Light and Cornerness structural-BoWs. 175
6.4 Average accuracies of structural-BoWs, Spatial-BoW and their combinations. Mo, Li, Co and spBoW correspond respectively to Motion, Light, Cornerness and Spatial BoWs. 180

7.1 Context Synopsis. 186
7.2 Average Accuracy of the different contexts on the HMDB dataset. . . 187
7.3 Average Accuracy of the different contexts on the UCF101 dataset. . 188
7.4 Context memory footprint. 190
7.5 Combination results. 191
7.6 Spearman's ρ factor. 192
7.7 Comparison with state-of-the-arts. Average Accuracy is reported. . . 193

8.1 Overview of the main thesis results. Average Accuracy is reported. . . 196

Chapter 1

Introduction

New social behaviors, societal transformations as well as the vast spreading of software systems have led to an explosion of data creation. The processing of Big Data, which denotes techniques to capture, process and analyze potentially large datasets, has become a major topic in the information technology field as it means new business opportunities, but also major research challenges [137]. According to McKinsey & Co, Big Data is "the next frontier for innovation, competition and productivity" [122]. In that event, Big Data has been flagged in Europe Horizon 2020 Strategy as a major target for research and innovation [29].

Multimedia content makes no exception to the Big Data trend. With the camera embedding in mobile phones, recent years have witnessed an explosion of multimedia data. Cameras are ubiquitous nowadays. More than 4 billion people or 60 percent of the world's population use mobile phones, and about 12 percent of those people have camera equipped smartphones, whose penetration is growing at more than 20 percent a year [122]. Consequently, the amount of visual content (images and videos) daily generated is overwhelming. In 2013 the video sharing website YouTube announced that 100 hours of videos were uploaded on its site every minute. They are watched by more than 1 billion individuals each month [51]. Albeit, the number of videos available has drastically increased this last decade, solutions for analyzing them in an automated fashion remain limited.

Computer vision systems, which are about automatically acquiring and interpreting the rich visual world around us, are still far behind the human vision abilities [86]. Search in large scale video databases still depends on costly manual annotation. Web search engines still rely on user provided textual descriptions to identify and retrieve multimedia data.

(a) Environment variability (b) Appearance variability

Figure 1-1: Double variability of computer vision

The core challenge of computer vision lies in a double variability [44] (see Figure 1-1). First, the visual content is subject to the environment variability, changes in illumination, viewpoint, occlusion and motion implies major transformations of the observed content. Yet, human vision can, without any difficulty, ignore those variations and reliably perceive the underlying materials. In addition, visual representations of materials also know strong variability: materials have multiple appearances that strongly differ. Chair objects, for instance, can take on many different forms. While humans are always able to recognize them as such, computers face more difficulty.

To leverage the astonishing growth of multimedia data produced, research in automated visual analyzing has never been more important. Vast collections of visual recordings remain a largely untapped resource because of the information extraction cost. By improving automated visual analysis systems, we could lower the cost of information extraction, leading to the development of a new generation of computer vision based applications.

1.1 Automated Multimedia Annotation

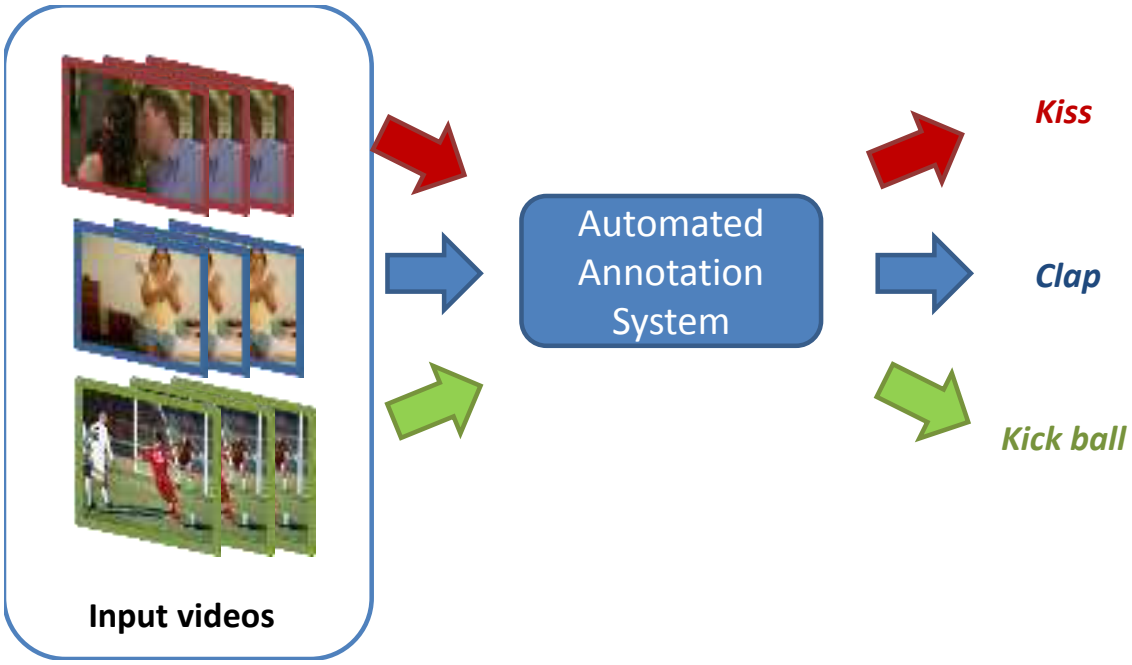


Figure 1-2: Video Annotation System.

Automated concept annotation, also called concept recognition, is at the core of the visual analysis problem. It aims at enriching visual data (photos or videos) with a textual description that highlights the data semantic content (see Figure 1-2). Annotations must be added in an automated way, without any human intervention, to deal with the large scale volume.

This dissertation tackles the problem of automated concept annotation in multimedia video. The video annotation problem is characterized by both the concept type (defining *what* we are looking for) and the video type (specifying *from where* we are looking for).

1.1.1 Concept Type

Different type of concepts can be detected in video. For instance, one can consider event, action, scene, object, activity, *etc.* No general agreement exists on those

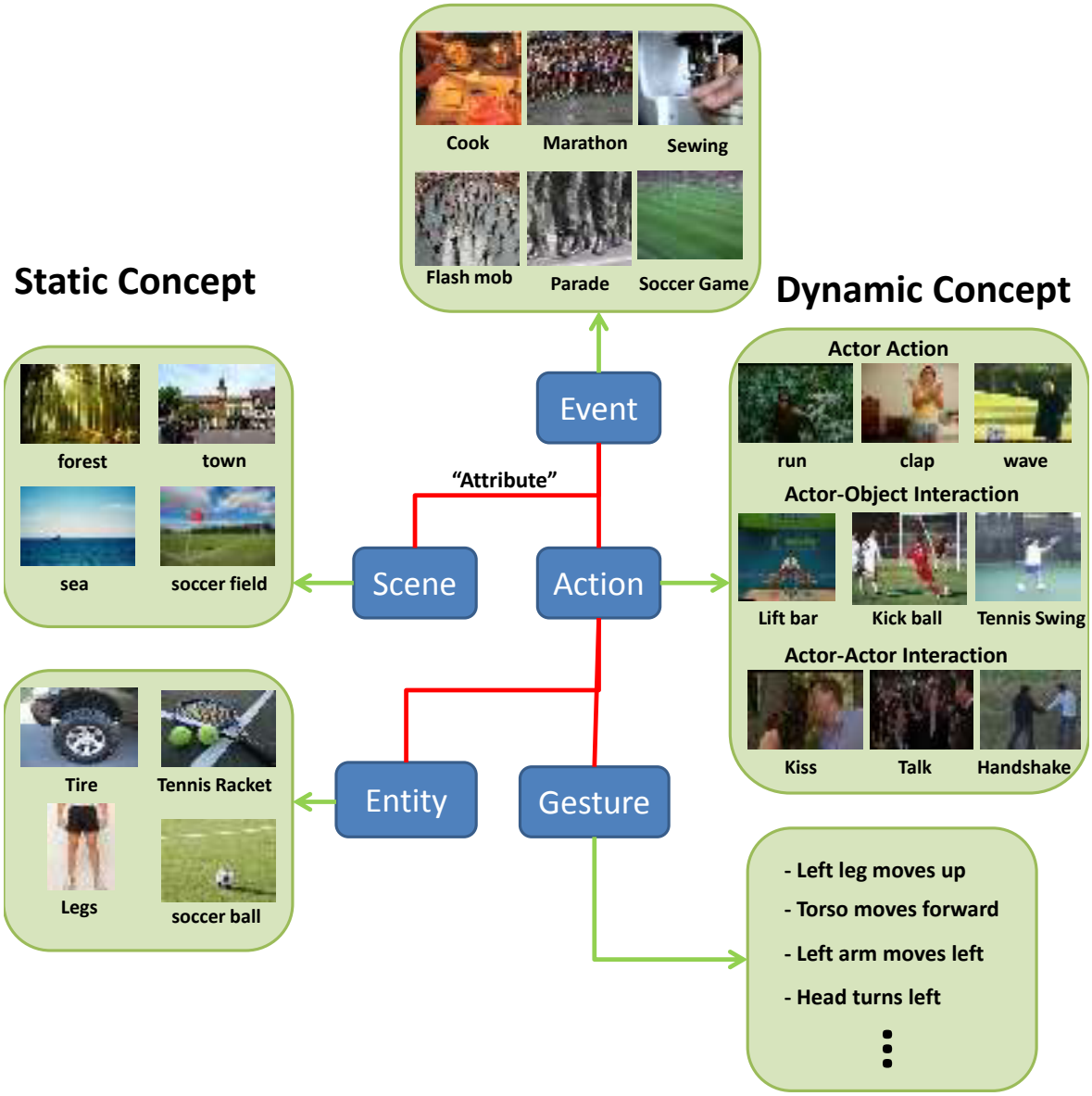


Figure 1-3: Concept Taxonomy.

terms precise definitions; they are being used interchangeably by the scientific literature [160].

Figure 1-3 proposes a hierarchical organization between different concept categories. It specifies the terminology used in the present document. At the bottom of the hierarchy, we find entity and gesture concepts. An entity is any objects (*soccer ball*, *tennis racket*) or actors, *i.e.* subjects accomplishing an act (*human*, *animal*), which compose a video. A gesture is a large displacement movement associated with an entity (*leg going up*). Next are action concepts, defined as a combination of gestures and entities achieving specific aims (*Kicking ball*, *Run*, *Hand shake*). As shown in Figure 1-3, three action sub-categories can be draw depending on the actor interaction with the environment (Actor, Actor-Object, Actor-Actor). Actions have a semantically meaningful interpretation, but span only on short temporal windows. By contrast events, defined as a sequence of actions, have large temporal duration (*Soccer Game*, *Marathon*). Events occur in a specific scene concept which captures the global environmental settings in which videos have been recorded (*soccer field*, *street*).

Figure 1-3 shows that actions have a central place in the concept hierarchy. Action concepts are semantically meaningful as they provide useful information which can be used to retrieve the underlying video data. In addition, there is a need for efficient action detectors as they provide basic building blocks that could be used to design event detectors. Events occur at a higher-semantic level in the concept hierarchy. Consequently, this thesis gives a particular attention on generic action recognition.

1.1.2 Video Type

The concept annotation problem is strongly impacted by the video data type. Video content is indeed very diverse and can be ordered in several categories as shown in Figure 1-4.

The video recording settings directly alter the concept appearance variability.

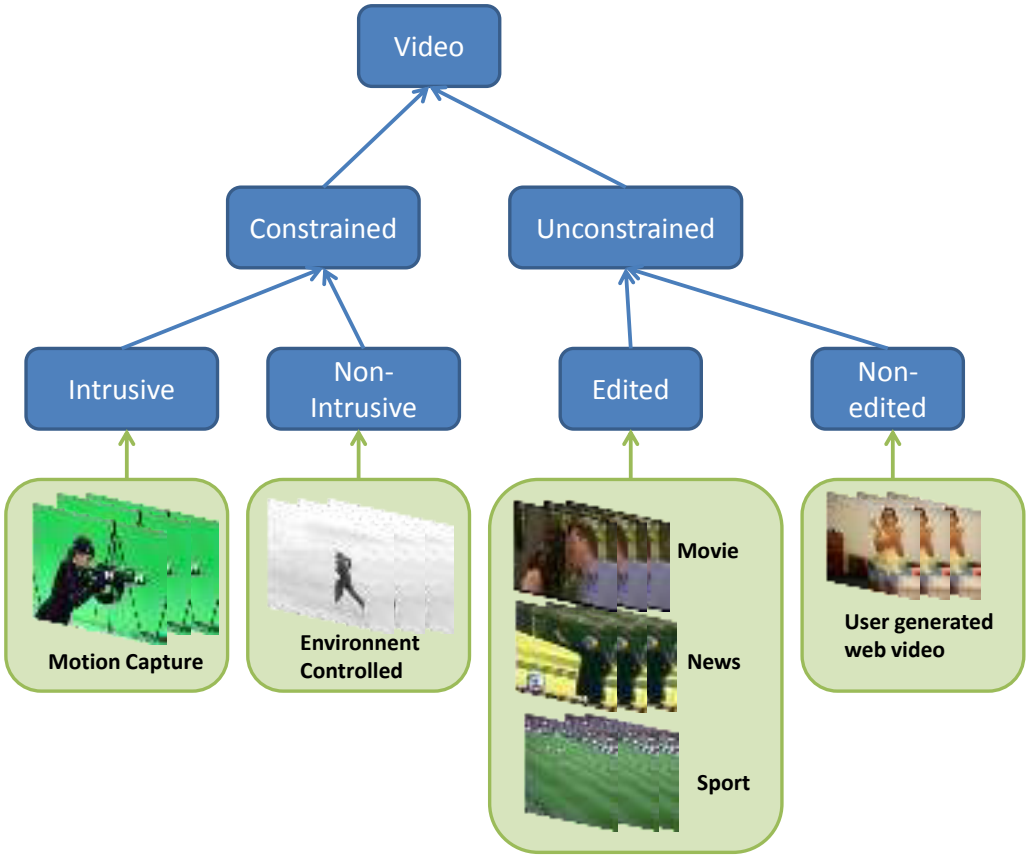


Figure 1-4: Video Taxonomy.

For instance, constrained videos are recorded in a carefully controlled environment which limits their visual complexity. Motion-capture videos, for instance, restrain the concept appearance variability by controlling the camera viewpoint and background clutter. In addition, motion-capture eases the video automated processing by adding intrusive markers attached to human actors that identify their main articulations. On the other side, no prior assumption can be made about unconstrained videos which are shot "in the wild". Such videos are subjects to strong visual variability due to camera viewpoint change , scene illumination variations, *etc.*

We are interested in handling data which are mostly user generated. By definition, we cannot make any prior assumption about the video except that the concept of interest, if present, is relatively well visible. Our concept-annotation solution has

to deal with unconstrained videos.

In this dissertation, we investigate automated concept annotation in unconstrained videos. While the developed approaches can be applied to event or action concepts, our experimental evaluation focuses particularly on the action level.

1.2 Thesis Main Contributions

Automated annotation systems challenge the concept recognition problem by first transforming the visual data into low-level signatures. Such signatures summarize the multimedia content key aspects by capturing its spatial and temporal patterns. The signatures are then exploited by statistical models which detect the presence of concepts. Although some improvements have been shown those last years (see Chapter 2), it still remains an unsolved problem, notably because of the strong variability inherent to the multimedia content.

In this thesis, we propose to enrich the low-level video representation of a video with multiple contextual information. Context is defined as "the set of circumstances in which a concept occurs". Any video signatures that capture appearance, motion or space-time information can be considered as contextual clues associated with a concept. We state that one context is not informative enough to discriminate a concept in a video. However, by considering several contexts at the same time, we can address the annotation problem. More precisely the thesis major contributions are the following:

- A new framework that takes into consideration several contextual information: To benefit from multiple contextual clues, we introduce a fusion scheme based on a generalized sparsity criteria. This fusion model automatically infers the set of relevant contexts for a given concept (Chapter 3).

- A feature covariance context: Different features capture complementary information. For instance, Histogram of Gradient (HoG) focuses on the video appearance while the Histogram of Flow (HoF) collects motion information [101]. Most of the existing works capture different feature statistics independently. By contrast, we leverage the local feature covariances to take advantage of the feature inter-dependencies (Chapter 4).

- A concept-dependent space-time context: Discriminative information is not equally distributed in the video space-time domain [102]. To identify the discriminative regions, we introduce a learning algorithm that determines the space-time shape associated to each individual concept (Chapter 5).

- An attention context: We leverage biological-inspired attention maps in video signatures. Such maps allow capturing space-time contextual information while preserving the video signature invariance to the translation, rotation and scaling transformations. Without this space-time invariance, different concept instances with various localizations in the space-time volume can lead to divergent representations. This problem is severe for the dynamic actions which have dramatic space-time variability (Chapter 6).

	State-of-art		Thesis	Gain
UCF-101 [178]	85.9	[178] 2012	**87.7**	8%
HMDB [95]	57.1	[70] 2013	**53.3**	-
UCF-50 [157]	84.5	[198] 2013	**92.7**	9%
UCF-Youtube [113]	84.0	[197] 2011	**86.3**	4%
KTH [162]	94.5	[49] 2011	**95.5**	1%
UT-Interaction 1 [160]	84.0	[144] 2012	**91.3**	9%
UT-Interaction 2 [160]	86.0	[144] 2012	**95.0**	11%

Table 1.1: Overview of the thesis results on publicly available datasets. Average Accuracy is reported.

The proposed contributions are extensively evaluated on several publicly available datasets (HMDB [95], UCF-50 [157], UCF-YouTube [113]...). As Table 1.1 shows,

we obtain competitive performances on those challenging datasets.

Chapter 2

Related Work and Experimental Datasets

This chapter proposes a survey of the multimedia annotation field.

Several surveys have already been proposed in this domain [1, 8, 80, 103, 151, 160, 175]. However, they tend to focus on one specific concept category. Aggarwal [1], Poppe [151] and Ryoo [160] provide a detailed description of video representation, classification models, and datasets used for human action recognition. Snoek [175] and Ballan [8] review approaches used in multimodal video indexing, with a particular interest for object entities and scenes concepts. Lavee [103] and Jiang [80] present some approaches developed for complex event analysis.

This chapter first proposes a global study of methods used for all the concepts types (entity, action, scene and event). To this end, we structure the existing works in two main categories:

- *Visual Data Representation* which contains intermediate representations that depict the multimedia content;
- *Concept Modeling* that proposes different approaches to capture the correlation between the intermediate representations and the concepts.

This chapter then presents a critical overview of the main experimental datasets

together with state-of-art performances. We conclude by highlighting the main bottlenecks of the existing works and the research direction explored in this thesis.

2.1 Visual Data Representation

This section describes some of the well-known representation used for multimedia data description. We limit our study to the visual and semantic features. For a detailed review of other multimodal features (audio, text) in multimedia, readers can refer to the survey of Atrey [3].

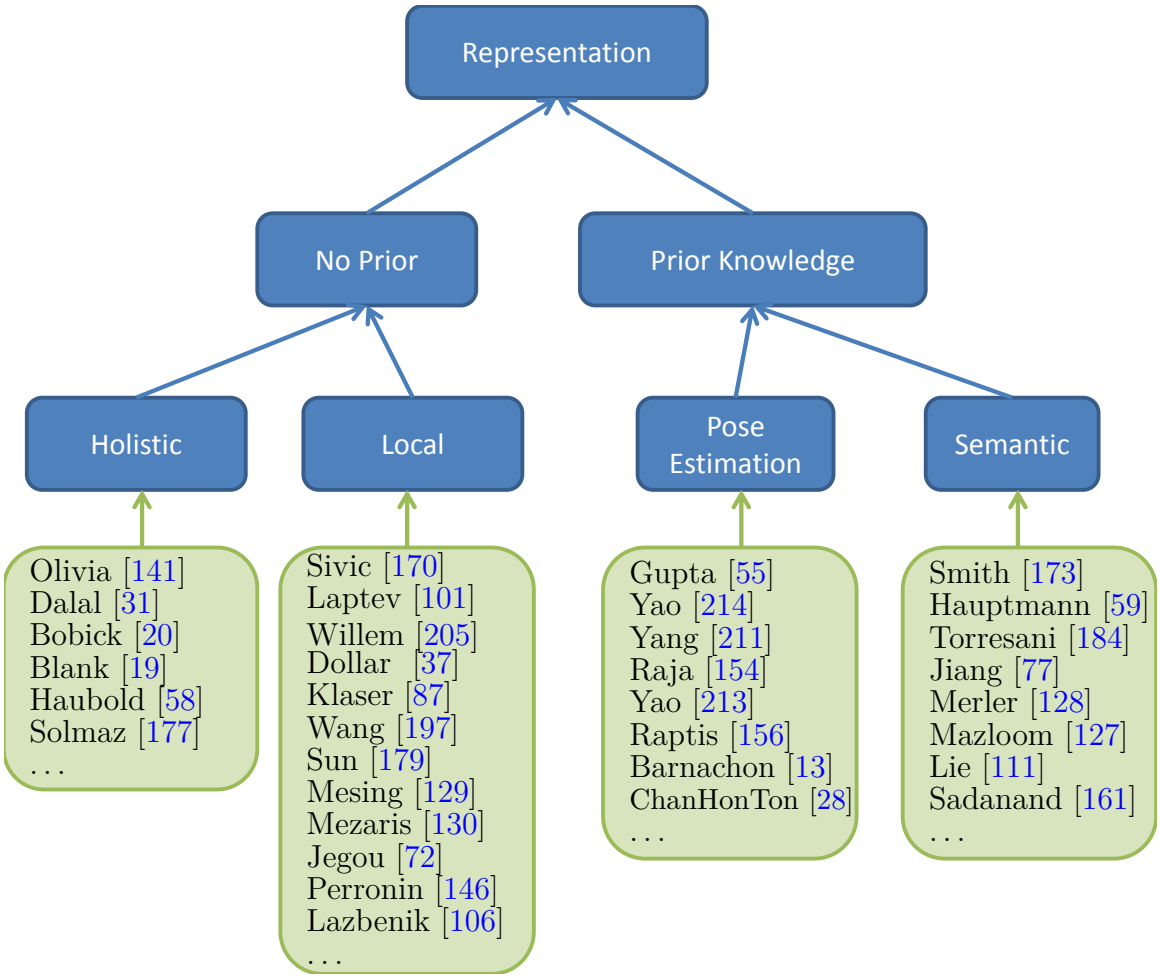


Figure 2-1: Taxonomy of image and video representations.

To structure the state-of-art survey, we propose in Figure 2-1 a taxonomy of the different visual representations. We first divide the representation in two main classes: representations that embed some prior-knowledge about the video or concept, and the representations with no-prior which are computed directly from the visual data. At the bottom of the taxonomy, we identify 4 visual representation categories: holistic, local, pose-estimation and semantic.

2.1.1 Holistic Representation

(a) MEI and MHI representations (courtesy of [20])

(b) 3D silhouette (courtesy of [19])

Figure 2-2: Examples of silouette based holistic representation.

Holistic representations [46, 138, 140] consider an image or a video as a whole. They depict multimedia content through the global distribution of low-level information (color [46, 216], texture [121, 140], shape [138], *etc.*).

Various low-level information can be considered, leading to different holistic representations (see Table 2.1). Popular image holistic representations include the ones proposed by Oliva and Torralba [141] and Dalal and Triggs [31]. Olivia *et al.* [141] introduce the GIST that encodes the dominant spatial structure of a scene. Dalal and Triggs [31] develop a Histogram of Oriented Gradient (HoG) which identifies and counts patterns of intensity gradient. In video, holistic approaches can rely on the human silhouette computed from background subtraction [19, 20, 52] (see Figure 2-2). However, such human-centric approaches are generally limited to constrained videos, the extraction of reliable silhouette features in realistic videos being already a challenging problem [52]. Video holistic representations can also characterize the motion

flow directly [32, 58, 177].

Method	Description
GIST [141]	Dominent 2D spatial structure
HoG [31]	Histogram of gradient intensity patterns
MEI and MHI [20]	2D silhouette descriptor based on motion substraction
3D Silhouette [19, 52]	3D silhouette volume descriptor
Motion image [58]	Sum global motion vector in an image
MbH [32]	Histogram of motion derivative patterns
GIST3D [177]	extension of GIST representation to 3D

Table 2.1: Holistics representation.

Although, holistic signatures have been shown suitable for concept recognition in unconstrained video data [177], they present certain drawbacks. Holistic representations are in general not invariant to viewpoint changes and camera motion. In addition, due to their global aspect, holistic representations are sensible to background clutter and occlusion. It needs to be counterbalanced. One approach would be to learn specific concept models for each particular view (frontal, lateral, rear, *etc.*) and environment setting (with or without occlusion, with or without camera motion,*etc.*).

2.1.2 Local Representation

Local representations have been introduced to provide visual signature robust to viewpoint change, background clutter and occlusion phenomena. A local representation aggregates the statistics of visual primitives, the local features, which tend to be stable under the previous phenomena.

Local features are image or video patterns that characterize given local neighborhoods. They are computed using (1) detectors that extract some image or video regions; (2) descriptors that characterize the information contained in the different regions. In particular, local feature detectors can focus on interesting point, *i.e.* sparse local regions computed with some criteria, or extract regions densely according to a

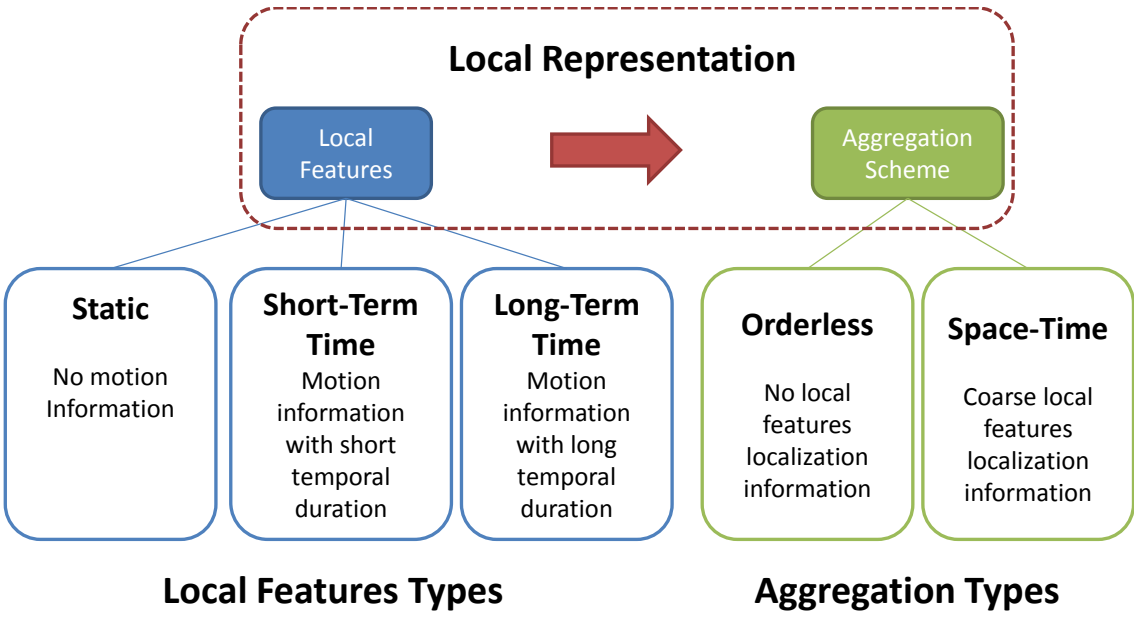


Figure 2-3: Local Representation Synopsis.

regular grid.

Once the local feature descriptors are computed, an aggregation scheme is applied to capture their associated statistics. Indeed, the number of local features extracted from visual contents is subject to variation. This variability poses difficulty in the image or video comparison since most similarity measurement requires fixed length inputs. One can address this problem by matching directly the local features between the different images or videos. However, the local-feature pairwise comparisons become quickly untractable when the dataset size augments, even with the help of indexing structure such as an inverted file system. Aggregation step solves this computational issue. Rather than matching exactly the local features of the different images or videos, it considers and compares descriptions of their statistical distributions. By relaxing the exact matching constraints, aggregation makes the image or video comparison tractable, even in presence of large scale datasets. In addition, statistical distributions improve the robustness of the representation.

As Figure 2-3 highlights, various local features and scheme exist in the literature. We review each different category in the following.

Static Local Features Static local features consider only visual appearance information extracted from image or video frames.

Lots of efforts have been dedicated to the design of 2D sparse detectors that extract distinctive regions from images [15, 57, 112, 116, 125, 132]. Popular region detectors include Harris [57], Hessian [112] and MSER [132] (see Table 2.2). While region detectors have been proven useful in the context of image matching, it has been observed that sampling features densely according to a regular grid (see Figure 2-4) leads to better recognition performances [139]. Although, it does not eliminate the need of sparse detectors. Tuytlaars *et al* [186] indeed demonstrate that combining both sparse detection and dense sampling offers the best performance in visual recognition tasks. As for detectors, different local feature descriptors have been investigated [61, 105, 115, 190] (see Table 2.2). The *Scale Invariant Feature Transform* descriptor (SIFT) [115] is the most popular amongst those descriptors.

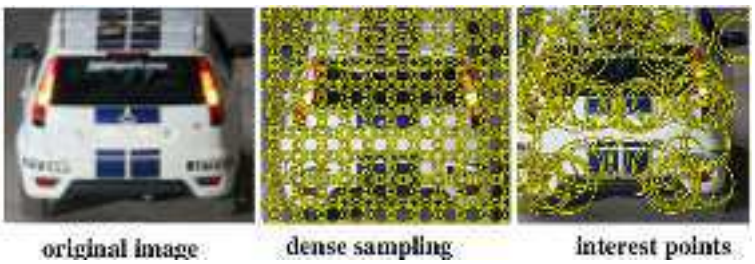


Figure 2-4: Static Features Sampling Strategies (coutesy of [186]).

Static local features have demonstrated state-of-art results for static image classification [38]. However such approaches do not take into consideration the temporal dimension which limits their abilities to discriminate videos.

Short-Term Time Local Features Short-term time local features, such as Space Time Interest Points (STIP) [100], have been introduced to leverage both appearance and motion information. STIP, for instance, extends the 2D Harris detector to the space-time domain by considering the temporal dimension as a third spatial dimension. STIP then describes the detected 3D regions using Histogram of Gradient (HoG)

Detector	
Harris [57]	Detect corner points based on the second moment matrix
Hessian [112]	Find regions using *Difference of Gaussian* filter
MSER [132]	Maximizes the size of connected components sharing
Dense Sampling [139]	Sample features according to a regular grid
Descriptor	
SIFT [115]	Distribution of intensity gradient orientation
ColorSIFT [190]	Extension of SIFT to color-space
RIFT [105]	Rotation invariant SIFT
CS-LBP [61]	Binarized Symmetric Intensity Pattern
SURF [14]	Computationally Efficient Descriptor

Table 2.2: Static Local Features.

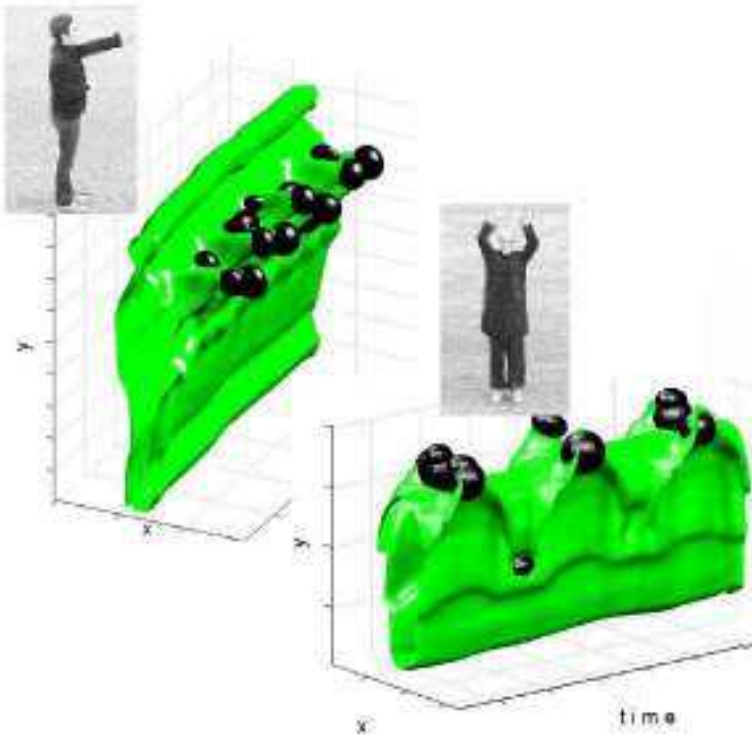


Figure 2-5: Space-Time Interest Point. Green zone correspond to the human silhouette, black zones are the detected salient regions (courtesy of [100]).

characterizing gradient pattern, and a Histogram of Flow (HoF) capturing the distribution of optical flow patterns [102]. Different variation of short-term time feature detectors and descriptor have been proposed in the literature [37, 87, 100, 102, 205] (see Table 2.3). Wang *et al.* [196] show that short-term time features detectors and hand-crafted descriptors have comparable performances.

Detector	
Harris3D [100]	Harris extension using space-time second moment matrix
Hessian [205]	3D extension of *Difference fo Gaussian*
Cuboid [37]	Gabor filters for 3D keypoint detection
Dense sample [196]	3D regular grids sampling
Descriptor	
HoG/HoF [100]	Histogram of Gradient and motion Flow
eSurf [205]	3D Surf extension
HoG3D [37]	Histogram of 3D gradient orirentations
STIPConv [107]	invariant descriptor learned with convolutional networks

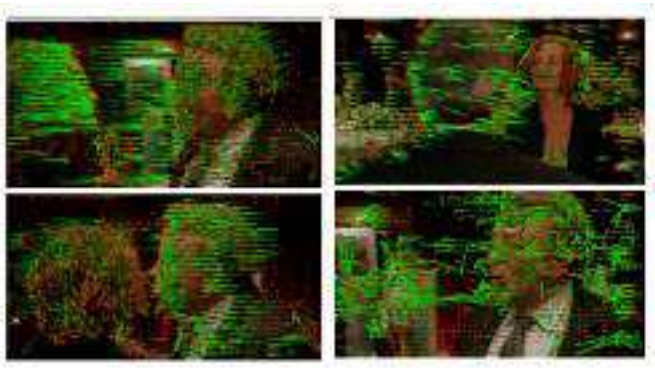
Table 2.3: Short-Term Time Features.

While taking into account the temporal dimension, these descriptors tend to be too localized in the space-time volume to characterize long term motion. Harris 3D detector [100] assumes that regions of interest know a rapid variation of motion (e.g. the regions motion are accelerating or decelerating). Dynamic actions and events can be characterized by motion patterns which don't contain sharp extrema in their variation [85]. Other local based representations have been investigated to cope with this issue.

Long-Term Time Local Features Trajectory features overcome the short temporal duration of STIP features. A trajectory is defined as a set of local regions found in successive frames which are constrained by space-time and visual appearance continuity [130]. By definition, trajectories capture long-term motion information in videos.

Trajectories are built by tracking 2D regions across the video frames. Several

(a) Sparse trajectories(courtesy of Sun [179])



(b) Semi-dense trajectories (courtesy of Wang [197])

Figure 2-6: Trajectory features.

Detector	
Trajectory contexts [179]	sparse SIFT features pairwise matching
KLT [129]	Sparse optical flow
Farneback [41]	Dense optical flow
Descriptor	
Hierachical contexts [179]	SIFT, motion-correlogram and trajectory-correlogram
HoGHoFMbH [197]	Histogram of Gradient, Flow and motion boundary
Velocity [129]	Derivative of motion vectors
Multiscale descriptor [130]	Multiscale Haar filter responses

Table 2.4: Long-Term Time Features.

tracking algorithm can be used (see Table 2.4). Farneback optical flow [41], extracting the trajectories densely, has been shown to outperform the other trajectory sparse sampling schemes [197]. Several descriptors can be used as well to encode the trajectory shape and motion information. In general, the combination several descriptors capturing different trajectory aspects (appearance, motion, velocity) augments the local descriptor discriminative power [9, 179, 197].

Motion descriptors of trajectory features are sensible to the camera motion. Recently some works [70, 81, 199, 207] have proposed to estimate the camera motion to counterbalance it in the motion trajectory description. Jain *et al.* [70], for instance, use a polynomial decomposition to separate the dominant from residual motion. They obtain state-of-art performance in several action recognition datasets.

Due to their awareness of the long-term motion context, trajectories have been shown to outperform both static and short-term features in the context of action and event recognition [198]. However, due to the tracking, computation of trajectory descriptors requires substantial computational overhead.

Orderless Aggregation Different aggregation schemes have been proposed in the literature [72, 146, 170]. The bag-of-words representation [170] (BoW) has been the most investigated representation by the community. In its traditional design [170], see Figure 2-7, a local feature codebook is constructed by quantizing local features extracted from a visual collection, using a k-mean clustering algorithm. Cluster centroids are the different words composing our codebook. They define spatial cells partitioning the local feature space. Given a new visual content, local features are extracted and associated to the index of their nearest words through hard-assignment. The distribution of the visual words in the visual content is captured through a histogram.

In practice, BoW performance is sensitive to many implementation choices [91]. BoW codebook computation can rely on generative [42], discriminative and sparse [104, 119] or kernel [191] approaches, leading to various degrees of performance improvement. An important finding is that associating a local feature to a sparse combination of visual words using some soft-assignment variation reduces the local feature quantization errors, and, significantly improves the BoW performance [114, 163, 210]. It has also been shown that the combination of multiple local features detectors and descriptors in the BoW representation also improves the classification performance [220].

Fisher Vector [146] is an alternative to the BoW aggregation relying on the fisher kernel principle. While BoW considers only the local features counting statistics, Fisher Vector goes beyond and captures up to the second order statistical information. It has demonstrated state-of-art performance on many datasets [146]. VLAD [72] is a fast-approximation of the Fisher Kernel.

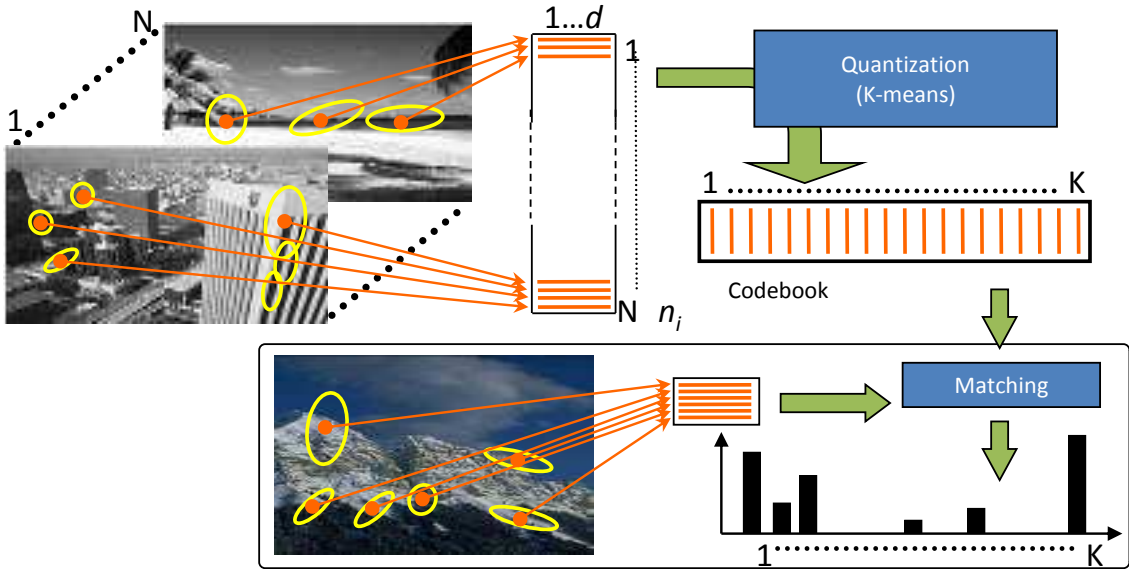


Figure 2-7: Synopsis of the Bag-of-Word model [170]. Local features are extracted from images, then quantized into a visual codebook. An image is then represented as a distribution of codebook words.

Space-Time Aggregation One major drawback of the BoW representation remains its lack of spatial information. BoW treats image and video as a collection of unordered elements; spatial localizations of local features are discarded in the representation which is not optimal since they convey discriminative information [93]. To address this issue, Lazebnik *et al.* introduce the Spatial Pyramid Matching [106]. They model coarsely the space-time information of image by partitioning a frame into rectangular grids at various levels, and computing a BoW histogram for each grid cell. Spatial Pyramid Matching has demonstrated state-of-art performances in the recognition task [106]. Laptev *et al.* [101] have proposed the Space-Time Grids which are the alter-ego of the Spatial Pyramid in videos. Space-Time Grids divide the space-time volume using predefined segmentation grids (see Figure 2-8) and also lead to performance improvement over BoW representation [102, 197]. Despite their

encouraging performance, Spatial Pyramid Matching and Space-Time are limited by fix geometry models which do not necessarily fit the spatial distribution of local features [56].

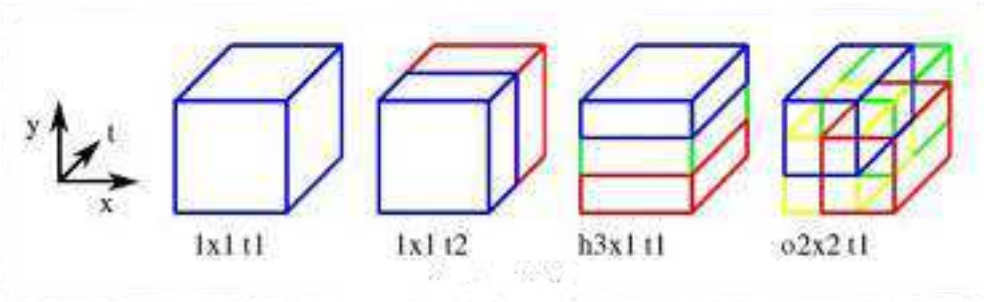


Figure 2-8: Exemple of 3D segmentation grid (courtesy of Laptev [100].)

A key advantage of local representations is their robustness toward viewpoint change, background clutter and occlusion phenomenon as well as their flexibility with respect to the video data. Local representations have been successful applied to unconstrained video data [197]. Despite those benefits, local representations generally have limited knowledge about the image or video global structure; they only provide a limited modeling of the local feature spatial distribution.

2.1.3 Pose-Estimation based Representation

Local representations have proven to be efficient for a variety of visual recognition tasks, but pixels or even local regions carry little semantic meanings. High level visual tasks could benefit from a more human-understable representation [111]. Pose estimation leverages the semantic associated with body pose (or human skeleton) localization. We know that a human skeleton (see Figure 2-9) captures rich and discriminative information since Johansson *et al.* [82]. They have demonstrated in the well-known moving light experiment that an observer recognizes a human action using only the motion associated with a few skeleton articulations.

Works have therefore investigated joint pose estimation and action recognition in still images [43, 55, 155, 211, 214] (see Table 2.5). They have shown that human pose

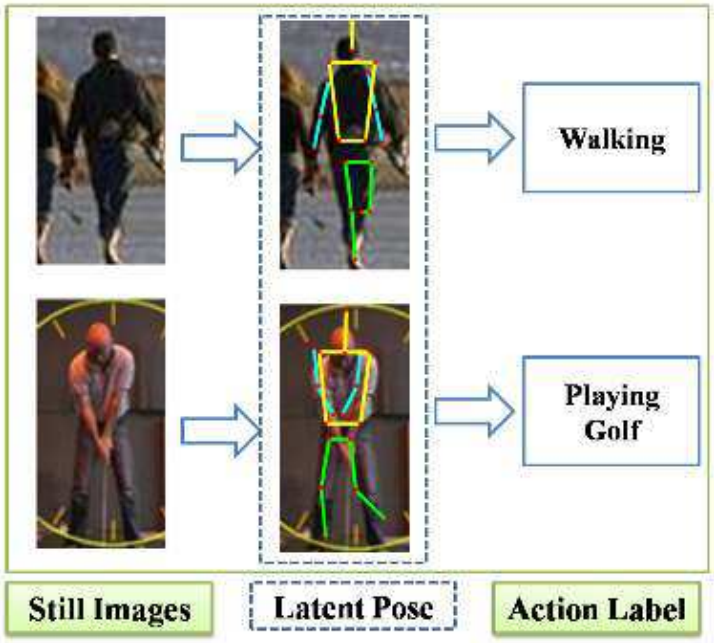


Figure 2-9: Pose Estimation for Action Recognition (courtesy of Yang [211]).

provides additional discriminative information useful for action recognition even in an unconstrained environment. However, such approaches typically require manual annotation of skeletons in the training dataset. It restrained their applicability since the number of available annotated training data is limited due to the high costs associated with the manual annotation. Raja *et al.* [154] have recently try to overcome this issue by propagating the annotation information in video using visual similarity. In addition, pose estimation also comes at a computational overhead price [150].

Method	Description
Bayesian Approach [55]	Graphical model for human-object interaction
Mutual Context [214]	Simultaneous pose and action estimation with random field
Latent Pose [211]	Simultaneous pose and action inferring with latent modeling
ImageGraph [154]	Skeleton training annotation Propagation
Combined Pose [213]	Combination of appearance and pose representations
Skeleton Corr [156]	Maximum normalized cross correlation of skeleton poses
ArticulationBoW [28]	Bag-of articulation trajectories

Table 2.5: Pose Estimation based representation.

Skeleton-based representations have also been investigated in videos. The recent

success in skeleton extraction, based on time-of-flight captor such as KINECT [131], has lead to works exploring human action analysis based on skeleton data [28, 156, 213] (see Table 2.5). As for static images, those works prove that using both skeleton and visual features improve the recognition performances. But, those approaches rely on time-of-flight captors which operate only in a strongly constrained environment.

Pose estimation has known a strong regain of success those last few years, notably due to the introduction of time-of-flight camera allowing a robust estimation of skeleton position in constrained environment. In addition, some recent works [211, 214] have shown the usefulness of pose estimation based representation for action recognition in realistic static images. The extensibility of those approaches to unconstrained video remains an open question.

2.1.4 Semantic Representations

While pose estimation is limited to human body information, semantic representation models the relation of various and generic concepts in multimedia content. Smith *et al.* [173] have defined the basis of semantic representation. They propose to build a vector space model by aggregating the confidence scores of independent concept models. It has been theoretical demonstrated by Hauptmann *et al.* [59] that a semantic representation based on fewer than 5000 concepts, detected with minimal accuracy of 10%, is likely to provide high accuracy results, comparable to text retrieval in a typical broadcast news collection.

Method	Description
Classemes *et al.* [184]	Weakly trained concept classifiers
DASD [77]	Domain adaptive semantic graph
SMV [128]	Semantic model vector based on ensemble-SVM
Informative Concepts [127]	Selection of Informative Concepts
Object Bank [111]	Scale-invariant concept detectors response map
Action Bank [161]	3D filter bank localizing semantic concepts in videos

Table 2.6: Pose Estimation based representation.

Given those observations, different works have investigated semantic representation [77, 127, 128, 184] (see Table 2.6). In particular, semantic approaches taking into account the concepts localization have demonstrated encouraging results for static image annotation [111]. However, an equivalent approach [161] applied to videos obtains only limited performances.

Semantic representation is a particularly interesting research direction since it allows adding some prior knowledge, captured by the concept semantic detector, in the visual representation. While having demonstrated state-of-art performance on image dataset, their extensibility to videos still needs to be demonstrated.

2.2 Concept Modeling

Machine learning algorithms are an important part of automated concept annotation systems. They learn the correlation between concepts and video intermediate representations. In this section, we detail machine learning algorithms used to detect the presence or absence of concepts in videos. In particular, we address 3 categories of concept modeling: linear and kernel methods, graphical models and information fusion.

2.2.1 Linear and Kernel Methods

Linear and Kernel-based classifiers have been popular in a wide range of applications for many years. Among many choices of kernel-based classifiers, Support Vector Machine (SVM) is the dominant paradigm for multimedia classification due to its reliable performance [7, 100, 101, 106, 115, 179, 194, 196]. In this section, we discuss several issues related to applying SVM to visual concept recognition. We start by considering the binary classification problem where we try to detect the presence of only one visual concept.

Binary Classification: Let's consider a labeled training dataset $\mathbf{X} = \{\mathbf{X}_i\}_{i\in[1,N]}$ where each $\mathbf{X}_i \in \mathbb{R}^{1\times D}$ is a video intermediate representation. We denote by $\mathbf{Y} = \{y_i\}_{i\in[1,N]}$ the binary label, $y_i \in \{+1, -1\}$, indicating the concept presence or absence. A SVM finds the hyperplane separating the positives from the negatives samples with the maximum margin. The margin is defined as the smallest distance between the hyperplane and training vectors. Given an unseen feature $\mathbf{X}_i$, a linear SVM computes its corresponding label through:

$$d(\mathbf{X}_i) = \mathbf{X}_i\mathbf{W} + b, \tag{2.1}$$

$\mathbf{W} \in \mathbb{R}^D$ is normal vector to the hyperplane and b is the model bias parameter such that $d(\mathbf{X}_i) > 0$ if $y_i = 1$ or $d(\mathbf{X}_i) < 0$ otherwise. $\mathbf{W}$ is expressed as a linear combination of the training vector: $\mathbf{W} = \sum_{i=1}^{N} \alpha_i y_i \mathbf{X}_i$, where $\alpha = \{\alpha_i\}_{i\in[1N]}$ are the Lagrange multipliers solving the following dual optimization problem:

$$\hat{\alpha} = \arg\max_{\alpha\geq 0} \sum_{i=1}^{N} \alpha_i - \frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N} \alpha_i\alpha_j y_i y_j x_i^T x_j. \tag{2.2}$$

The previously described SVM algorithm assumes that a linear separation exists between the two classes. It is usually not the case in realistic learning applications. SVM has been therefore extended to no-linear separation. Video representations are projected to a high dimensional space using a projection function ϕ. The optimal hyperplane is then computed in the high dimensional feature space by solving the following quadratic problem:

$$\hat{\alpha} = \arg\max_{\alpha\geq 0} \sum_{i=1}^{N} \alpha_i - \frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N} \alpha_i\alpha_j y_i y_j \phi(x_i)^T \phi(x_j) \tag{2.3}$$

Computing the inner product of vectors in a high dimensional space is computationally expensive. Kernel has been introduced to avoid this issue. A kernel function k is a function mapping pairs of feature vectors to real numbers. If the kernel function k respects the Mercer conditions: continuous, symmetric and positive semi-definite, then $k(x_i, x_j)$ expresses an inner product in high-dimensional space: $k(x_i, x_j) = \phi(x_i)\phi(x_j)$.

The performance of SVM classification is sensitive to a few parameters, the most critical one being the kernel function choice. The selection of a suitable kernel depends on the input vector data distribution, which varies from task to task. Zhang *et al.* [221] propose a comparison of linear, RBF, EMD-based and χ^2 kernels for BoW image representation. This study shows that χ^2 and EMD generally outperform the other kernels. Yang *et al.* [210] show that one can reach the best classification with a linear kernel by modifying the BoW design. Specifically they demonstrate that the combination of local feature sparse coding and linear kernel achieves better performance than hard assignment and χ^2 kernel. Keeping a linear kernel is critical for large scale application. While the χ^2, and EMD implies a quadratic learning complexity, linear kernel learning complexity remains linear.

Given the proliferation of visual signatures and kernels, some methods have been developed to combine them. Multiple Kernel Learning (MKL), which has been introduce by Bach *et al.* [6], learns the optimal kernel combination from the training data. Gönen *et al.* [50] and Bucak *et al.* [24] provide MKL reviews for computer vision. They show that, considering non-linear kernels, MKL performs better than unweighted or data-dependent combination. However, MKL combination is only equivalent to feature weighting in the context of linear kernels. It does not outperform unweighted or data-dependent combination. Using kernelized SVM implies a computational overhead which limit its usability with large scale dataset.

Multiple Concepts Classification Realistic classification problems contain more than two concepts to recognize. Several strategies have been introduced to extend the binary classification problem to multiclass classification.

One of the simplest strategies is to train one-vs-rest binary classifiers for each class, using all available training vectors. Multiclass SVMs have also been proposed. Weston *et al.* [204] introduce a multiclass SVM with a loss function that leverages each class-wise losses. Lee *et al.* [109] and Crammer *et al.* [30] describe another multiclass SVM, statistically consistent [2], which apply the multinomial classification idea to the "hinge-loss" function. Albeit the many multiclass SVM extensions, Akata *et*

al. [2] show through an extensive experimental study that the simple one-vs-rest strategy outperforms all the other method in term of performance and computational efficiency, for visual recognition.

Due to their performances and efficiency, SVMs classifiers are predominant in the multimedia annotation field. However, SVM doesn't provide a direct solution to combine multiple intermediate representations. MKL has been introduced tackle this issue, but they lack of performance gain when linear kernels, necessary to handle large data scale, are involved.

2.2.2 Graphical model

A Graphical model encodes the conditional relationship of a set of random variables, in a form of a graph, leading to compact representations of probabilistic distributions. Directed graphical models also known as Bayesian networks [135] (BN) were the first used to model the concepts semantic relation. Let $\mathbf{C} = \{c_j\}_{j\in[1,M]}$ be a set of concept and $\mathbf{X} = \{\mathbf{X}_i\}_{i\in[1,N]}$ a set of video representations. To completely specify a Bayesian network, two sets of parameters need to be defined: $P(c_j|\mathbf{X}_i)$, the first layer capturing the video representation and concept correlation and $P(c_j|c_{j'})$, the second layer describing the concept co-occurrence statistics (see Figure 2-10). The graph model can either be complete or sparse , using ontologies [34, 159] or learned from a training dataset [60, 76, 152, 203, 208, 209], to improve the computation time. One drawback of Bayesian models is the lack of temporal modeling. Only spatial co-occurrences are studied. Dynamic Bayesian networks (DBN) [62, 193] address this issue by fusing both temporal and spatial dimensions in the graphical models. Dynamic Bayesian networks are a generalization of HMM, directly modeling the temporal concept dependencies. It leads to complex models depending on a large number of parameters.

Bayesian networks and dynamic Bayesian networks are generative probabilistic frameworks leveraging the joint representation and concept probability: $P(c_j, \mathbf{X}_i)$ which requires the representation inter-relation modeling. However, since it is difficult to model complex relations of the observed data while retaining computational

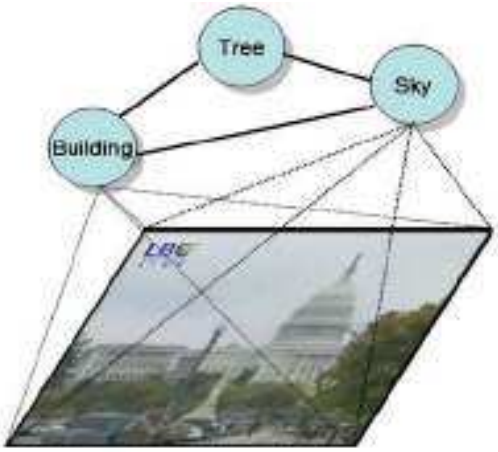


Figure 2-10: Two layers undirected graphical model (courtesy of Hauptmann [60]).

tractability, generative approaches assume the independence of each observed features [97, 181]. This assumption is too restrictive for computer vision [96].

On the other hand, discriminative approaches describe the concepts posterior probability $P(c_j|\mathbf{X}_i)$ and don't require the features relationship modeling. Following this idea, undirected graphical models, such as Conditional Random Fields (CRF) [142], or their 2D extension Discriminative Random Fields (DRF) [96] have been introduced. It has been shown that CRF outperforms Bayesian models in classification task at price of a costlier learning phase [96, 97, 142, 209].

Graphical models build a factorized representation recognizing visual concepts from low-level representation. These models provide an implicit level of abstraction in understanding concept relationship which can provide valuable insight. Although the approaches discussed under this section are mathematically and computationally elegant their success in realistic recognition problem is still inconclusive [80].

2.2.3 Information Fusion

Information fusion deals with systems that have information sources available. By using a proper combination scheme, fusion aims at decreasing the influence of unreliable sources compared to the reliable ones [89]. The fusion of different information sources can be performed at different levels: representation, decision or "hy-

brid" [3, 33, 99, 189].

The representation level, also called early fusion, combines directly the low-level video representations [176]. Early fusion presents the advantage of using the low-level representation inter-correlation. However, it is often difficult to associate different low-level video signatures into a common representation. Each low-level representation comes from a different feature space which is distributed accodingly to some specific underlying statistics. It may not be mixable, without proper normalization, with other feature spaces [89]. In addition, early fusion augments the dimensionality of the video signature by combining several representations, increasing the learning complexity. The decision level, or late fusion, first applies classifiers on each extracted representation and obtains intermediate decisions scores. These scores are combined together, in a fused representation [176]. Late fusion doesn't take advantage of the low-level representation inter-correlation. But, this fusion scheme is more flexible than early fusion since a dedicated classifier can be designed for each input representation. Furthermore, decisions scores share the same representation which eases their combination. As shown by Snoek *et al.*[176], there is no consensus about which fusion scheme gets better performance. Their efficiencies depend on the low-level representations and on the data distribution. Hybrid level [99] consists in combining the two levels of fusion together, low-level features and classifier scores, trying to take advantage of both early and late fusion.

There is more to the fusion design than the choice of the fusion level. We also need to specify the fusion method, which defines how to combine the different information. Linear weighted fusion is the approach generally adopted in multimedia annotation [78, 79, 176]. It associates a specific weighting coefficient to each input information source. Linear coefficient can be determined using various approaches. A straightforward approach is to set equal coefficient to all input information source [68]. Albeit its simplicity; this method has shown reliable performance in complex event detection [78, 79]. Other approaches use cross-validation to determine the optimal weight associated to each source [65, 136]. Multiple Kernel Learning [179] can be considered as a fusion method that learns the weight coefficient from a training dataset.

2.3 Experimental Datasets

Several standard video datasets have been proposed by the community to evaluate and compare the concept annotation approaches [95, 102, 113, 157, 160, 162, 171, 178]. Datasets have different scale and complexity as summarized in Table 2.7. Two broad categories can be draw from them: constrained and unconstrained datasets.

Constrained datasets are recorded by the scientists directly in a controlled environment. The initial human action datasets (KTH [162], Weizman [52]...) were taped in a supervised environment in order to control the video complexity. Since constrained datasets are built by researchers, they generally contain a limited number of videos. Such datasets are particularly useful for highlighting a recognition algorithm particular aspect. However, algorithms achieving good results on constrained dataset are not guaranteed to generalize well on unconstrained data.

While first recognition approaches were evaluated on constrained datasets, research community has largely shifted its attention toward realistic and unconstrained datasets [95, 113, 123, 157, 171, 178]. Such datasets are constructed from existing videos such as users generated web videos [113, 157, 178] or professionally edited videos [95, 123], *i.e.* movies or tv news... Consequently, those datasets don't control the recording environment. They are composed by videos which are generally subject to strong appearance variability due to viewpoint change, camera motion, background clutter... In the following, we describe the datasets used in this dissertation which are showed in Table 2.8.

2.3.1 UT-interaction Datasets

UT-Interaction [160] (see Figure 2-11) is actually composed by two sub-datasets UT-1 and UT-2. Each sub-dataset has 6 classes of human-human interaction actions: *hands-shake*, *point*, *hug*, *push*, *kick* and *punch*. UT-1 and UT-2 are recorded in constrained environment. UT-1 is composed by 60 videos occurring on a parking lot. The videos are taken with different zoom rates and with mostly static backgrounds.

Dataset	Concepts	Videos	Viewpoint Change	Motion	Clutter
UT-Inter [160]	6	120	None	Weak	Weak
KTH [162]	6	2392	None	Weak	None
CUHA [180]	14	68	None	None	None
TUM [183]	19	1000	None	None	None
UCF-Youtube [113]	11	1668	Strong	Strong	Medium
Hollywood2 [123]	13	1684	Strong	Strong	Strong
UCF-50 [157]	50	6681	Strong	Strong	Strong
UCF-101 [178]	101	13320	Strong	Strong	Strong
HMDB [95]	51	6849	Strong	Strong	Strong
Trecvid SIN 2012 [171]	362	8000	Strong	Strong	Strong
Trecvid MED 2012 [171]	20	40000	Strong	Strong	Strong

Table 2.7: Datasets overview in term of Concepts number, Videos Number, Viewpoint Change, Camera motion and Background clutter.

	Action Type			Video Type		
Dataset	Human	Human-Object	Human-Human	Constrained	Web	Movie
UT-Inter [160]			√	√		
KTH [162]	√			√		
UCF-Youtube [113]	√	√			√	
UCF-50 [157]	√	√			√	
UCF-101 [178]	√	√	√		√	
HMDB [95]	√	√	√		√	√

Table 2.8: Action Type for Human-Action Datasets.

Kick Punch Hug Handshake

Figure 2-11: Frame samples from the UT-interaction datasets.

Authors	Descriptions	UT-1	UT-2
Ryoo *et al.* [160]	STIP	85	75
Dollar *et al.* [160]	Cubois/HoGHoF	85	75
Patron *et al.*[144]	Head Pose-Estimation	84	86
Chapter 5	Dense trajectories + Adaptative Grid Pooling	**91.7**	**95**

Table 2.9: Results on UT-interaction.

UT-2 is composed by the 60 remaining videos. The UT-2 videos occur in a park and have non static backgrounds. They are also subject to small camera jitter. The evaluation procedure is specified by Ryoo [160], it uses a 10-fold leave-one-out cross-validation on segmented video shots. Average accuracy is reported for each action class.

Table 2.9 reports the performances of various methods obtained on the UT-Interaction datasets. Being Human-Human interaction, UT-interaction actions see their localizations change through time in a video. By taking into account a flexible space-time context (see Chapter 5), we achieves state-of-art performance on this dataset.

2.3.2 KTH Dataset

KTH [162] (see Figure 2-13) is another constrained dataset. KTH is composed by 6 human action classes: *Boxing*, *Handclapping*, *Handwaving*, *Jogging*, *Running*, *Walking*. Each action class is performed several times by 25 subjects. The videos were recorded in four different scenarios: outdoors, outdoors with different zoom rates (to induce scale variation), outdoors with different clothes, and indoors. The videos

Figure 2-12: Frame samples from the KTH dataset.

Authors	Descriptions	Results
Klaser *et al.* [87]	Harris3D/HoG3D	84.3
Dollar *et al.* reported in [196]	Cubois/HoGHoF	88.7
Laptev *et al.* reported in [196]	Harris3D/HoGHoF	91.6
Shi *et al.* [167]	Dense cuboid sampling + HoGHoFMbHHoG3D	93.0
Wang *et al.* [197]	Dense trajectories+HoGHoFMbH	94.2
Kovashka *et al.* [92]	Hierarchical Vocabulary	94.5
Gilbert *et al.* [48]	Hierarchical data mining	94.8
Chapter 6	Dense trajectories + Content based Pooling	94.6
Chapter 4	Dense trajectories + Covariance Pooling	**95.5**

Table 2.10: Results on KTH.

are mostly non-cluttered static backgrounds. Evaluation is performed using a training/testing division provided in [162].

Table 2.10 reports state-of-art results. One special feature of this dataset is the high similarity between its *Jogging* and *Running* action. By proposing an aggregation method which goes beyond the BoW counting statistics, we are able to improve over the state-of-art (Chapter 4).

2.3.3 UCF-Youtube, UCF-50, UCF-101 Datasets

UCF-Youtube, UCF-50 and UCF-101 are three unconstrained datasets composed by user generated videos uploaded on the YouTube website. Videos contained in the three datasets are therefore subject to high-appearance variability, large camera motion, viewpoint change, cluttered backgrounds... The YouTube dataset [113] is composed by 1168 video sequences distributed in 11 different actions: *shooting (basket)*, *biking*, *diving*, *swinging*, *swinging (golf)*, *swinging (tennis)*, *jumping (tram-*

Figure 2-13: Frame samples from the UCF datasets.

poline), *spiking (volleyball)*, *horse riding*, *walking* and *juggling (soccer)*. UCF-50 [157] extends the YouTube dataset to 50 different human actions and 6681 video sequences also extracted from the YouTube website. Finally, UCF-101 proposes 51 additional actions, reaching the total of 101 actions and 13320 videos. To our knowledge, UCF-101 is the largest video dataset available. In the literature a 25 folds leave-one-out group-wise crossvalidation is generally used for evaluation

Many video signatures have been evaluated on those datasets as Table 2.11, 2.12 and 2.13 shows. Capturing long-term time information, Bag-of-Word based on dense trajectories has shown particularly encouraging performance. We improve upon this representation by adding structural information to a traditional BoW representation, as Chapter 5 and 6 describe, and achieve state-of-art performance.

Authors	Descriptions	Results
Liu *et al.*[113]	Mined 2D SIFT and motion features	71.2
Ikizler *et al.*[64]	Gist + object and person centric HoGHoF	75.21
Wang *et al.*[197]	Dense trajectoires + HoGHoFMbH	84
Chapter 5	Dense trajectories + Adaptative Grid Pooling	86.3

Table 2.11: Results on UCF-Youtube.

Authors	Descriptions	Results
Klipper-Gross *et al.* [88]	Motion Interchange Pattern	72.6
Solmaz *et al.* [177]	GIST 3D	73.7
Reddy *et al.* [157]	Scene and motion descriptor late fusion	76.9
Shi *et al.* [167]	Dense cuboid sampling + HoGHoFMbHHoG3D	83.3
Wang *et al.* [197]	Dense trajectories+HoGHoFMbH	84.5
Chapter 4	Dense trajectories + Content based Pooling	92.8

Table 2.12: Results on UCF50.

Authors	Descriptions	Results
Soomro *et al.* [178]	Harris3D/HoGHoF	44.5
Chapter 7	Multiple-Contexts	**87.7**

Table 2.13: Results on UCF101.

2.3.4 HMDB Dataset

HMDB [95] (see Figure 2-14) is composed by 6849 video clips divided into 51 action categories. They are collected from various sources, mostly from movies, and public websites. It contains simple facial actions, general body movements, human-object interaction and human-human interactions. Videos are subject to strong difference in their recording condition as Figure 2-15 highlights. Camera motion, various viewpoint and video quality are available for each action. It contains simple facial actions, general body movements and human interactions. [95].

Dense trajectories feature also have good results on this dataset. As for the UCF-datasets, adding structural information in the representation also improve the results (see Table 2.14).

Figure 2-14: Frame samples from the HMDB dataset.

Authors	Descriptions	Results
Kuehne *et al.* [95]	C2	23.0
Sadanand *et al.* [161]	Action Bank	26.9
Cao *et al.* [26]	STIP+temporal pooling	27.8
Klipper-Gross *et al.* [88]	Motion Interchange Pattern	29.2
Solmaz *et al.* [177]	GIST 3D	29.2
Jiang *et al.* [81]	Dense trajectories+HoGHoFMbH+motion compensation	40.7
Wang *et al.* [197]	Dense trajectories+HoGHoFMbH	46.6
Shi *et al.* [167]	Dense cuboid sampling + HoGHoFMbHHoG3D	47.6
Jain *et al.* [70]	Dense trajectories+HoGHoFMbH+motion compensation	52.1
Wang *et al.* [198]	Dense trajectories+HoGHoFMbH+motion compensation	**57.1**
Chapter 5	Dense trajectories + Adaptative Grid	46.8
Chapter 4	Dense trajectories + Covariant Pooling	51.1
Chapter 6	Dense trajectories + Content based Pooling	51.8
Chapter 7	Mutiple-Contexts	**53.6**

Table 2.14: Results on HMDB.

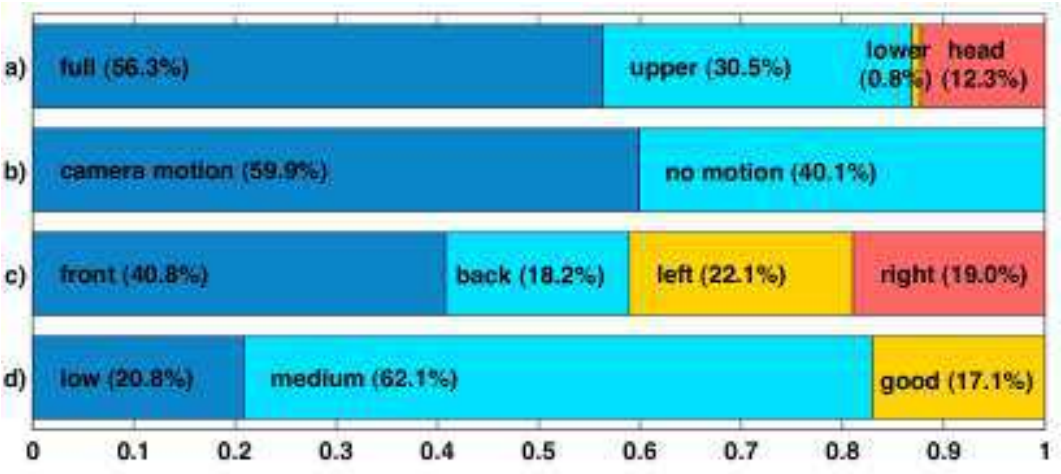


Figure 2-15: Distribution of the various conditions for the HMDB videos (courtesy of Kuehne [95]). a) visible body part, b) camera motion, c) camera view point, and d) clip quality.

2.4 Conclusion

To conclude this chapter, we identify some state-of-art bottlenecks for both video representations and concept models. Based on this limitation, we highlight the different research directions which are investigated in this dissertation.

2.4.1 Video Representations

Representation	Advantage(s)	Drawback(s)
Holistic	Computational Efficiency	Sensible background change and motion
Local	Flexible, Robust, Performance	Lack of semantic, structural information
Pose	Meaningful, Performance	Limited to constrained data
Semantic	Meaningful, additional knowledge	Requires large training data

Table 2.15: Synopsis of the video representations.

Table 2.15 summarizes advantage and inconvenient of each type of video representation. We are interested in actions recognition for unconstrained videos. We need a representation tackling video data with large appearance variability in order to detect concepts with relatively simple semantic meanings.

Due to its flexibility and robustness, local representation is a good fit to depict unconstrained videos. Local representation is not exempt of disadvantages as it tends to lack from structural information. In particular, we identify three drawbacks which, if tackled, could refine the representation discriminative capability:

- **Bag-of-Words Higher Order Statistics:** Local representation, using BoW aggregation, tends to focus on local descriptor first order statistics. They don't explicitly consider the descriptor co-variations. Descriptor covariance has a potentially strong discriminative ability. It is especially relevant to action representation since covariance describes mid-level patterns that characterize jointly the motion and appearance in video, while at the same time an action is defined jointly by a specific movement and appearance. We therefore investigate a covariance context to refine the video representation in Chapter 4.

- **Task-Specific Space-time Information:** Space-time information conveys discriminative information [106]. However, most of local representation approaches have a limited modeling of the video space-time context. State-of-art solutions embed local feature space-time information in a bag-of-words model through statically defined segmentation grids. Such approaches use the same segmentation layout for all the actions. Consequently, there is no guarantee that the segmentation grids will fit the local feature space-time distribution. To tackle this issue, we explore, in Chapter 5, an action-specific space-time context that learns action-adapted segmentation grids directly from the video data.
- **Space-time Modeling and Invariance Trade-off:** Local representations that capture space-time information, lose the space-time invariance. They are not robust to global transformations in the space-time domain. We state that being invariant to space-time transformations is of primary importance in unconstrained videos. Actions are indeed subject to strong space-time localization variations in videos. We therefore propose in Chapter 6 a new representation that leverages space-time context while being robust toward global space-time transformations. Our approach relies on attenttion map estimated using saliency functions.

2.4.2 Concept Modeling

Linear and Kernel-based approaches have demonstrated high-performance in the automated annotation task [80]. In addition, linear classifier are scalable to large data thanks to their limited training complexity. We therefore choose to rely on linear classifers to detect the presence of concepts in video.

One limitation of linear approaches remain their lack of multiple representations modeling. Multiple Kernel Learning (MKL) addresses this issue. However, it shows limited performances when combined with linear models [50].

To tackle this issue, we investigate the embedding of sparse contraints in the classification framework. Such idea has been originally proposed for various learning

problem [5]. In particular, we study group-sparsity regularization for a linear SVM based classification (see Chapter 3). To this end, we take advantage the group sparsity constraints introduced by Ma [212], expressed with a $\|.\|_{2,1}$ norm in the context of image reconstruction. We adapt the group sparsity constraints to a squared hinge loss function. We also study the impact of group-sparse $\|.\|_{2,p}$ $(p < 2)$ in order to have a finer control on the sparsity selection.

Following, Ma [212] we adopt a block-coordinate descent to optimize our learning problem. However, other optimizations such as proximal approaches can be considered for such problem. An extensive review and comparaison of the different optimization methods applied with sparsity regularization has been proposed by Bach *et al.* [4]

Chapter 3

Contribution: A Contextual View of Video Annotation

This chapter introduces a general framework for automated video annotation. Our framework relies on two major observations: (i) multi-contextual description is necessary to capture the video content richness and diversity; (ii) some contexts are more informative about the presence of a concept in multimedia content than others. Using this insight, we propose a learning framework that automatically determines which are the relevant contexts associated with a concept.

In the remaining of this chapter, we start by introducing the framework motivations. We then define the notion of context from a high-level point-of-view. Finally, we formulate a video annotation framework which automatically selects the most significant contexts given a concept.

3.1 Motivation

Multimedia videos are extremely rich representations that aggregate visual, audio and textual signals. Recent years have witnessed an explosion of multimedia contents available. As chapter 1 highlights, the video sharing website YouTube announced in 2013 that 100 hours of videos, which approximately correspond to 480 million

books, were uploaded on its site every minute [51]. Considering the astonishing data volumes, a human analysis of each video is no longer feasible. The need of automated visual analysis has never been more crucial.

Definition 1. *Concept: a fundamental category of existence which is used to denote a class of things in the world*

Concept annotation addresses the visual analysis problem. It consists in mapping the abundant flow of visual information to human-understandable abstractions. Rather than using their low-level information, we want to characterize multimedia data through small textual descriptions that synopsize their key aspects. Automated concept-annotation raises the semantic gap problem [172].

Definition 2. *Semantic gap: Lack of correlation between a high-level human understanding of a visual content and its low-level representation.*

Figure 3-1: Example of *car* image under different viewpoint and illumination parameters.

The semantic gap results from the divergence between two representations of multimedia content: the concepts, human-understandable high-level representations, and the signals, computational low-level representations. This contradiction is induced by numerous real world physical phenomena such as photometry change, viewpoint change, object deformation or dynamic camera motion... Those phenomena generally imply an important variations in the visual appearance of a multimedia content, and so in its recording, without actually changing its semantic meaning. While human-understandable representation remains static under those phenomena, a low-level representation is likely to be subject to strong variations. As a result, two

low-level representations, noticeably different, can actually correspond to the identical high-level representation. For instance, Figure 3-1 shows different images having strong variations in their appearances, but, the concept of interest *car* remains the same in all of those images.

To bridge the gap, researchers have investigated the use of intermediate visual representations [9, 102, 115, 130, 170, 194, 196, 197]. Those representations aim at being invariant to the semantic gap related transformations. For instance Lowe [115] and Sivic [170] design an image signature aiming to be insensitive to rotation, translation and scale transformations to achieve viewpoint robustness. Benefiting from the robustness, the correlation between intermediate and high-level representation should be more apparent than the correlation of low-level signal and high-level representation.

In practice, a clear trade-off appears within an intermediate representation: the more invariant an intermediate representation is, the less discriminative power it will have. If we design an intermediate representation invariant to the geometric transformations (translation, rotation, scale), it will achieve robustness toward variability implied by viewpoint changes. This property is desirable to recognize global scene characteristics (*forest*, *urban*, *house*...) which are generally depicted through a myriad of point of views. On the other hand, a geometric invariant representation will also lose the geometric organization information about the visual data that can be useful to discriminate some concepts such as rigid objects (*car*, *bike*...) While an invariance property can be benefiting for a concept, it can also remove some discriminative information characterizing another concept.

Definition 3. *Context: the set of circumstances in which a concept occurs.*

Because of its inherent trade-off, one intermediate representation is not sufficient to describe a visual content. It will capture information that is either too specific or too general for some concepts. To tackle this issue, we propose to (i) enrich the

representation of a multimedia content with multiple contextual cues; (ii) learn what are the discriminative contexts associated with each concept.

Each visual intermediate representation can be considered as a specific context. A context characterizes a specific aspect of the multimedia signal. Context is not limited to the visual, it can also be extracted from the text or audio signals. We state that one context is not informative enough to discriminate a concept in a video. However, by considering several contexts at the same time, we can address the semantic gap.

The underlying assumption is that the optimal trade-off between intermediate representation invariance and discrimination is concept dependent. Some concepts, like global scene characteristics, know high appearance variability. In this case, highly invariant representations are more efficient. Differently, some concepts need intermediate representations that retain more discriminative information. Based on this hypothesis, we propose to automatically infer the optimal intermediate representations associated with a concept.

To extract a human-understandable textual description of multimedia video data, our framework first needs to extract multiple context signatures which characterize our content. Relying on those contextual information, we then model the correlation between the intermediate signatures and the high-level concepts.

3.2 Context

To specify our video annotation framework, we define from an abstract perspective the notion of context.

Definition 4. *Let* $\mathbb{V} = \{V_1, V_2, ..., V_N\}$ *be a set of videos. A context signature is defined as a real valued fixed length vector* $\mathbf{X} \in \mathbb{R}^{1\times D}$ *representing a multimedia content* V*. D is the dimensionality of the contextual space. We define as context extractor function, the function* $f : \mathbb{V} \rightarrow \mathbb{R}^{1\times D}$ *that maps a multimedia content to a context signature* $\mathbf{X} = f(V)$*.*

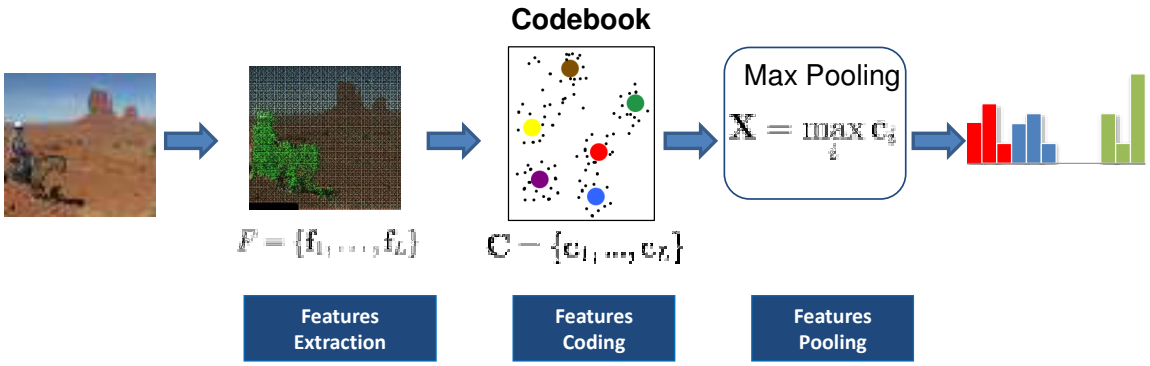


Figure 3-2: Illustration of a Bag-of-Words context.

Our context definition is intentionally broad to model the information diversity of multimedia data. In this thesis, we mainly consider information extracted from visual aspect of the video. Contexts are therefore equivalent to visual signatures in the following. Bag-of-Words (BoW) is an example of visual context (see Figure 3-2). However, our model can easily be extended to other type of contextual information such as uploader tags, user characteristics or video GPS coordinates. Related to this thesis, the use of video textual contexts has been investigated in [149].

Despite the representation diversities, we identify three majors categories of visual contexts feature, space-time and semantic. Each context category focuses on a particular aspect of the video information.

- Feature contexts are video signatures which characterize the video visual and audio signals. Such contexts capture information related to the video appearance, motion or audio. They allow the leverage of the low-level video signal information in the automated annotation framework.

- Space-time contexts are defined as *any space-time information that encapsulates the space-time layout and transition, relative position, global and semi-local statistics etc, of the low-level visual features* [69, 168, 179]. Space-time contexts model the geometric layout of the videos. But, while embedding discriminative geometric information, most space-time contexts lose of the geometric invariance. We identify two sub-levels of space-time level: spatial and temporal

which characterize the direct geometric relationships between visual features in the video volumes.

- Semantic contexts describe a video in term of high-level concepts. Concepts do not happen in isolation, and, the concepts semantic relations can be used in their detections [135]. For instance the detection of the concepts *shark* and *desert* in a same shot seems unlikely, while the detection of *car* should increase the probability of seeing *road*. Moreover, psychophysics studies [12, 18] have shown that human biological vision doesn't rely exclusively on appearance, but is complemented by the analysis of semantic relationship. The semantic context modeling could therefore help for automated concept annotation. Semantic context signatures are either "sensory", extracted from the video content, or "non-sensory", provided by third part resources such as tags, user description or other meta-data....

	Features Contexts			Space-Time Contexts		Semantic Contexts
	App	Motion	Audio	Spatial	Temporal	
SIFT-BoW [170]	√					
SPM [106]	√			√		
STIP-BoW [100]	√	√		√		
Traj-BoW [196]	√	√		√		
Scene Pooling [26]	√	√			√	
Augmented BoW [17]	√	√			√	
MFCC [75]			√			
Mined features [48]	√			√		
SIN346 [11]						√
Object Bank [111]						√
Action Bank [161]						√
Multi-Pronged [60]				√	√	√

Table 3.1: Taxonomy of existing methods in term of context categories.

A context is not necessary exclusive to one category, it can addresses the modeling of several multimedia aspects. Table 3.1 classifies using our context categories some of the existing works which have been performed in multimedia annotation those

last years. It shows that our terminology covers the large spectrum of the different existing works. In addition, Table 3.1 highlights that the scientific community has been less implicated in the investigation of some context. From our knowledge, only Cao [26] and Bettadapura [17] consider the temporal context for multimedia annotation. Multimedia representation could benefit from the modeling of such context.

3.3 Developed Framework

Defining the correlation between video context signatures and high-level concepts lies at the heart of the automated concept annotation. Knowing the correspondence between the two representations, we can infer the value of unobserved concepts given observed signatures. To estimate this correlation, we adopt a data-centric approach through machine learning (also called statistical modeling). Machine learning relies on an annotated training dataset to determine the dependencies between several variables, the intermediate and high-level representations in our case.

3.3.1 Model

Figure 3-3 presents a global overview of our concept annotation framework. Our framework considers several videos as input. It extracts multiple contexts from those videos, leading to several intermediate representations. Association between the different contexts and a high-level concept is then captured by our concept model (see Section 3.3.3)). Our model selects relevant contexts through group sparsity criteria (see Section 3.3.4)). In the following, we start by formalizing the automated concept annotation problem.

3.3.2 Problem Formulation

We consider the binary classification problem. Our aim is to detect the presence or absence of a concept in a set of videos.

Let $\mathbf{D} = \{\mathbf{V}, \mathbf{Y}\}$ be a training dataset composed by N videos $\mathbf{V} = \{\mathbf{V}_1, ..., \mathbf{V}_N\}$.

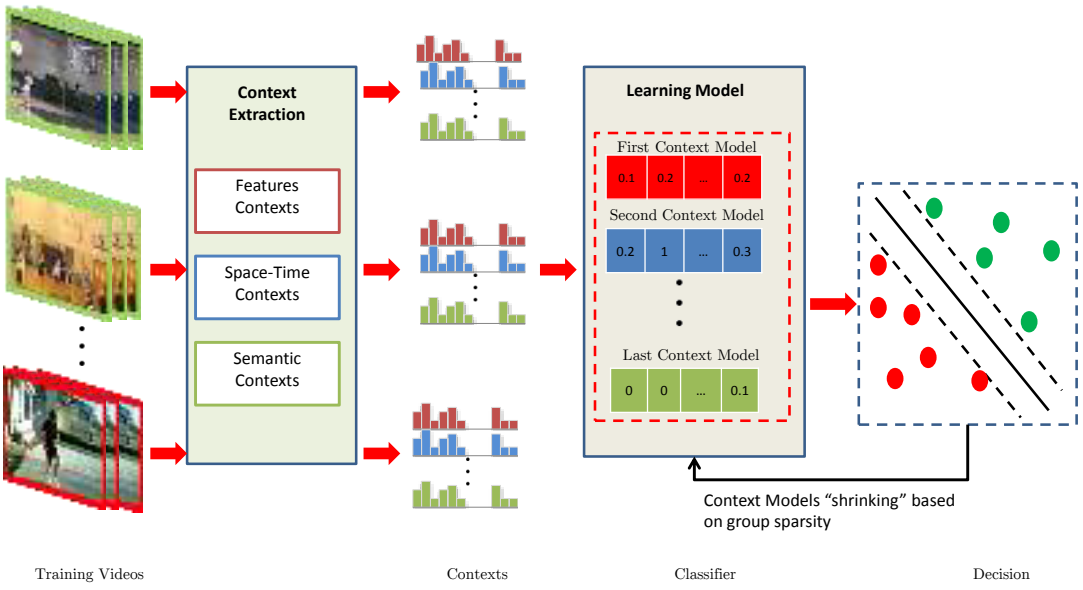


Figure 3-3: Framework Synopsis.

$\mathbf{Y} \in \{0,1\}^N$ are the video binary labels which indicate the presence or absence of the concept. We consider a set of C different context extractors $\{f_1, ..., f_C\}$ leading to N contextual signatures $\mathbf{X} = \{\mathbf{X}_1, ..., \mathbf{X}_N\}$. Each video signature $\mathbf{X}_i$ is the concatenation of C contexts, *i.e.* $\mathbf{X}_i = [\mathbf{X}_i^1, \ldots, \mathbf{X}_i^C]$ where $\mathbf{X}_i^c \in \mathbb{R}^{1 \times D_c}$ is the c-th context of the i-th video.

Concept annotation requires addressing two problems:

1. we need to *learn* a model which captures the correlation between the videos signatures $\mathbf{X}$ and labels $\mathbf{Y}$;
2. we need to *infer* the most likely annotation of new video signatures having unobserved labels.

3.3.3 Energy Based Modeling

To solve problem (1) and (2), we adopt the energy-based formalism. Energy-based modeling [108] measures the compatibilities between two configurations of variables through an energy function E. $E(\mathbf{X}_i, Y_i)$ can be interpreted as the degree of agree-

ment between our signature $\mathbf{X}_i$ and the label Y_i. High-value values of E expresses incompatible configuration between $\mathbf{X}_i$ and Y_i while small values of E correspond to compatible configurations.

Assuming the energy function is known, solving the *inference* problem consists in finding the label $\hat{Y}_i$ that minimize the energy function E given $\mathbf{X}_i$,

$$\hat{Y}_i = \underset{Y \in \{0,1\}}{\arg\min}\, E(\mathbf{X}_i, Y). \tag{3.1}$$

Alternatively, the *learning* problem requires to determine the energy function E that "fits at best the training dataset $\mathbf{D}$". We introduce $\mathbf{W}$, a set of parameters which characterize the energy function E. We denote by Γ the space of the different parameter values, *i.e.* $\mathbf{W} \in \Gamma$. Our goal is to find the $\mathbf{W}$ value such as E "fit at best $\mathbf{D}$". The "fit at best" criterion is evaluated through a loss functional (3.6), also called objective function,

$$O(\mathbf{W}, \mathbf{D}) = \sum_{i=1}^{N} L(Y_i, E(\mathbf{W}, \hat{Y}_i, \mathbf{X}_i)) + \lambda \Omega(\mathbf{W}). \tag{3.2}$$

In the objective function, $\hat{Y}_i$ is the predicted label from $\mathbf{X}_i$ which minimizes the current energy function: $\hat{Y}_i = \arg\min_{Y \in \{0,1\}} E(\mathbf{W}, \mathbf{X}_i, Y)$. L is a loss function that penalizes incorrect prediction ($\hat{Y}_i \neq Y_i$). Ω is the regularizing term that constraints energy function E complexity. Intuitively, this can be seen as an application of the Occam'razor [108]. In practice, regularizer allows to avoid the learning of energy functions that overfit the finite training dataset $\mathbf{D}$. Here, λ is a trade-off parameter between the empirical risk term ($\sum_{i=1}^{N} L(Y_i, E(\mathbf{W}, \hat{Y}_i, \mathbf{X}_i))$ and the regularization penalization term ($\Omega(\mathbf{W})$). *Learning* therefore consists in minimizing the empirical risks with a regularization penalty,

$$\hat{\mathbf{W}} = \underset{\mathbf{W} \in \Gamma}{\arg\min}\, O(\mathbf{W}, \mathbf{D}). \tag{3.3}$$

To complete our framework definition, we need to specify the loss function, the reg-

ularizer and the energy function. In the context of a binary classification problem and an energy function is generally defined as $E(\mathbf{W}, \mathbf{X}_i, Y_i) = Y_i M(\mathbf{X}_i, \mathbf{W})$ where $M(\mathbf{X}_i, \mathbf{W})$ is a classification model [108]. The choice of M, loss function L and the regularizer Ω is dependent on the classification problem,

$$L(\mathbf{Y}_i, E(\mathbf{W}, \mathbf{X}_i, \hat{Y}_i)) = \max(0, E(\mathbf{W}, \mathbf{X}_i, Y_i)) - E(\mathbf{W}, \mathbf{X}_i, \hat{Y}_i)) \quad (3.4)$$

$$= \max(0, 1 - 2Y_i M(\mathbf{X}_i, \mathbf{W})). \quad (3.5)$$

For the multimedia annotation task, Linear Support Machine (LSVM) is a very popular choice [2, 210]. Such approach specifies M as a linear model, $M(\mathbf{X}_i, \mathbf{W}) = \mathbf{X}_i\mathbf{W}+b$ where b is the model bias, Ω as a ℓ_2 regularizer, $\Omega(W) = \|\mathbf{W}\|_2$ and L as a square hinge loss (6.15).

3.3.4 From Multiple Contexts to Concept: Generalized Sparsity Regularization

Traditional energy-based learning framework (3.6) considers all the different contexts equally through one energy function. Contexts have different discriminative powers depending on the concept to recognize. While video motion information is primordial to distinguish concepts which have close appearances but different dynamics ("Running" or "Walking"), it does not characterize well rigid objects which are not subject to motion ("Chair" or "Table"). We therefore propose to learn the relevant contexts associated to a concept by constraining our energy function E. By focusing only on a few contexts, we could take advantage of intermediate representations which describe at best the concept of interest while discarding irrelevant and noisy signatures.

To explicit the use of several contexts in the energy-based modeling (3.3), we decompose $\mathbf{W}$ in a set of coefficient groups $\mathbf{W} = [\mathbf{W}_1, ..., \mathbf{W}_C]$. $\mathbf{W}_c$ is a parameters vector of the model which correlates with the c-th context. Rather than using one

model for all contexts, we rewrite (3.3) as a combination of C different models,

$$\begin{aligned} O(\mathbf{W},\mathbf{D}) &= \sum_{i=1}^{N} L(Y_i, \sum_{c=1}^{C} E_c(\mathbf{W}_c, \hat{Y}_i, \mathbf{X}_i^c)) + \lambda\Omega(\mathbf{W}) \\ &= \sum_{i=1}^{N} L(Y_i, \hat{Y}_i \sum_{c=1}^{C} M_c(\mathbf{W}_c, \mathbf{X}_i^c)) + \lambda\Omega(\mathbf{W}). \end{aligned} \tag{3.6}$$

(3.6) provides more flexibity than (3.3) since it allows to design one model per context. However, it does not express any restriction on the different contexts. All contexts are equally weighted using this model. In practice, contexts are not equally effective to represent a concept, and we aim at selecting only the most discriminative contexts while discarding the irrelevant ones. We add sparsity constraints to $\mathbf{W}$ through the regularizer in order to embed these structural constraints of our energy function E.

Traditional learning approach relies on a $\|.\|_2^2$ as regularizer [210]. $\|.\|_2$ norm attaches the same importance to each coefficient in $\mathbf{W}$, *i.e.*, each group $\mathbf{W}_c$ contributes equally. Sparsity is generally induced through the use of a $\|.\|_p$ norm with $p < 2$. However, this method implicitly assumes that each individual coefficient in $\mathbf{W}$ is independent of all the others. It only guarantees sparsity at the $\mathbf{W}$ individual coefficient level and does not assure that a few groups $\mathbf{W}_c$ will be selected by our learning framework. Group sparsity, on the other hand, uses a $\|.\|_{2,p}$ norm, a combination of a $\|.\|_p$ norm at the groups level and a $\|.\|_2$ norm at the individual coefficient level. While selecting only a few contexts with the $\|.\|_p$ norm, it considers the coefficient associated to a context $\mathbf{W}_c$ a whole through the $\|.\|_2$, taking advantage of their implicit relation. Hence, a $\|.\|_{2,p}$ regularization term is used in our learning formulation (3.7), reducing the number of selected contexts,

$$O(\mathbf{W},\mathbf{D}) = \sum_{i=1}^{N} L(Y_i, \hat{Y}_i \sum_{c=1}^{C} M_c(\mathbf{W_c}, \mathbf{X}_i^c)) + \lambda||\mathbf{W}||_{2,p}^p. \tag{3.7}$$

In (3.7), p controls the group selection sparsity. The smaller p is, the fewer groups are selected by the model. If $p = 2$, we obtain a classic ℓ_2 regularizer term. In this sense, (3.7) generalizes the traditional learning framework with ℓ_2 regularization.

3.3.5 Optimization

To *learn* an energy function E adapted to the training dataset, we need to minimize our loss functional $O(\mathbf{W}, \mathbf{D})$ with respect to $\mathbf{W}$. Assuming the convexity and smoothness of the loss funcion L, traditional energy-based learning, *i.e.* with a ℓ_2 regularizer, is a convex and smooth optimization problem. However, the use of sparse $\|.\|_{2,p}$ regularizer implies the lost of the smoothness property if $p \geq 1$ ((3.7) is not twice differentiable anymore). Moreover, we loose the convexity property if $p < 1$. To overcome this issue, we adopt an iterative relaxation strategy to optimize (3.7).

$$\hat{\mathbf{W}} = \arg\min_{\mathbf{W}\in\Gamma} O(\mathbf{W}, \mathbf{D}) \tag{3.8}$$

$$\Leftrightarrow \hat{\mathbf{W}} = \arg\min_{\mathbf{W}\in\Gamma} \sum_i L(Y_i, \hat{Y}_i \sum_{c=1}^{C} M_c(\mathbf{W_c}, \mathbf{X}_i^c)) + \lambda ||\mathbf{W}||_{2,p}^p.$$

$$\Leftrightarrow \hat{\mathbf{W}} = \arg\min_{\mathbf{W}\in\Gamma} \sum_i L(Y_i, \hat{Y}_i \sum_{c=1}^{C} M_c(\mathbf{W_c}, \mathbf{X}_i^c) + \lambda \sum_{g=1}^{G} \frac{\|\mathbf{W}_g\|_2^2}{\frac{2}{p}\|\mathbf{W}_g\|_2^{2-p}}. \tag{3.9}$$

We reformulate our objective function as (3.9). To relax our problem, we introduce a diagonal block matrix $\mathbf{D}$ defined as[1]:

$$\mathbf{D} = \begin{pmatrix} (\frac{2}{p}\|\mathbf{W}_1\|_2^{2-p})\mathbf{I}_1 & & \\ & \ddots & \\ & & (\frac{2}{p}\|\mathbf{W}_G\|_2^{2-p})\mathbf{I}_G \end{pmatrix}. \tag{3.10}$$

$\mathbf{D}$ is a semi-definite positive matrix. $\mathbf{I}_g$ is the identity matrix corresponding to the group $\mathbf{W}_g$. We observe that:

$$(\sum_{g=1}^{G} \|\mathbf{W}_g\|_2^p) = tr(\mathbf{W}^T\mathbf{D}^{-1}\mathbf{W}) = \|\mathbf{U}^T\mathbf{W}\|_2^2, \tag{3.11}$$

where $\mathbf{U}^T$ is the $\mathbf{D}^{-1}$ Cholesky decomposition ($\mathbf{D}^{-1} = \mathbf{U}\mathbf{U}^T$). We can therefore rewrite our optimization problem (3.9) as (3.12).

$$\hat{\mathbf{W}} = \arg\min_{\mathbf{W}\in\Gamma} \sum_i L(Y_i, \hat{Y}_i \sum_{c=1}^{C} M_c(\mathbf{W_c}, \mathbf{X}_i^c) + \lambda tr(\mathbf{W}^T\mathbf{D}^{-1}\mathbf{W}) \tag{3.12}$$

[1] In practice we add a ϵ to each diagonal coefficient of $\mathbf{D}$ for numerical stability.

Algorithm 1 Coordinate Descent.

Input: Signatures $\mathbf{X} \in \mathbb{R}^{N \times d}$ and labels $\mathbf{Y} \in \{0,1\}^N$. Regularization parameters λ, p
Output: $\mathbf{W}$

1: Initialize $\mathbf{W}$ at random;
2: **repeat**
3: $\quad \mathbf{D} = \begin{pmatrix} (\frac{2}{p}\|\mathbf{W}_1\|_2^{2-p})\mathbf{I}_1 & & \\ & \ddots & \\ & & (\frac{2}{p}\|\mathbf{W}_G\|_2^{2-p})\mathbf{I}_G \end{pmatrix}$
4: $\quad \mathbf{W} \leftarrow \arg\min_{\mathbf{W}} \sum_i L(Y_i, \hat{Y}_i \sum_{c=1}^{C} M_c(\mathbf{W_c}, \mathbf{X}_i^c) + \lambda tr(\mathbf{W}^T\mathbf{D}^{-1}\mathbf{W})$;
5: **until** Convergence

By fixing $\mathbf{D}$, we now obtain a convex and smooth optimization problem (3.12), assuming L is convex and smooth. However, $\mathbf{D}$ is an unknown variable which is dependent on $\mathbf{W}$ that also needs to be determined. We therefore use a coordinate descent procedure to optimize jointly $\mathbf{D}$ and $\mathbf{W}$ in algorithm 1.

It should be noticed that if $p < 1$, we still lose the convexity property of our loss functional. The optimization algorithm then converges to a local optimum.

3.3.6 Proof of Error Convergence

In the following, we demonstrate the error convergence of the energy-based learning framework with group sparse regularizer. We denote by $\mathbf{W}$ the optimal E parameters for the t-th iteration and $\mathbf{W}^*$ the result of algorithm 1 at $(t+1)$th iteration.

Lemma 1. *The following inequality holds*

$$\lambda \sum_{g=1}^{G} \|\mathbf{W^*}_g\|_2^p - \lambda \sum_{g=1}^{G} \frac{\|\mathbf{W^*}_g\|_2^2}{\frac{2}{p}\|\mathbf{W}_g\|_2^{2-p}} \leq \lambda \sum_{g=1}^{G} \|\mathbf{W}_g\|_2^p - \lambda \sum_{g=1}^{G} \frac{\|\mathbf{W}_g\|_2^2}{\frac{2}{p}\|\mathbf{W}_g\|_2^{2-p}}. \tag{3.13}$$

Proof. We consider the function $f(x) = \frac{1}{2-p}(2x - px^{\frac{2}{p}}) - 1$. We have $f(x) \leq 0 \; \forall x > 0$ Therefore by setting $x = \frac{\|\mathbf{W^*}_g\|_2^p}{\|\mathbf{W}_g\|_2^p}$ we obtain

$$\frac{1}{2-p}\left(\frac{\|\mathbf{W^*}_g\|_2^p}{\|\mathbf{W}_g\|_2^p} - p\frac{\|\mathbf{W^*}_g\|_2^2}{\|\mathbf{W}_g\|_2^2}\right) - 1 \leq 0. \tag{3.14}$$

By multiplying each side of (3.14) with $(1-\frac{1}{\frac{2}{p}})\lambda\|\mathbf{W}_g\|_2^p$ we obtain

$$\lambda(\|\mathbf{W^*}_g\|_2^p) - \lambda\frac{\|\mathbf{W^*}_g\|_2^2}{\frac{2}{p}\|\mathbf{W}_g\|_2^{2-p}} \leq \lambda(\|\mathbf{W}_g\|_2^p) - \lambda\frac{\|\mathbf{W}_g\|_2^2}{\frac{2}{p}}. \tag{3.15}$$

Hence, by extension

$$\lambda(\|\mathbf{W^*}_g\|_2^p) - \lambda\frac{\|\mathbf{W^*}_g\|_2^2}{\frac{2}{p}\|\mathbf{W}_g\|_2^{2-p}} \leq \lambda(\|\mathbf{W}_g\|_2^p) - \lambda\frac{\|\mathbf{W}_g\|_2^2}{\frac{2}{p}\|\mathbf{W}_g\|_2^{2-p}}. \tag{3.16}$$

By summing (3.16) over all the groups, we obtain our inequality (3.13).

□

Theorem 1. *Assuming an algorithm exists to solve* $\arg\min_{\mathbf{W}} \sum_i L(Y_i, \hat{Y}_i \sum_{c=1}^{C} M_c(\mathbf{W_c}, \mathbf{X}_i^c) + \lambda tr(\mathbf{W}^T\mathbf{D}^{-1}\mathbf{W})$*, the objective function (3.12) is iteratively decreased by Algorithm 1.*

Proof. For each iteration, we can solve the objective function (3.12) for a fixed value of $\mathbf{D}$. It results that:

$$\begin{aligned}\sum_i L(Y_i, \hat{Y}_i \sum_{c=1}^{C} M_c(\mathbf{W_c^*}, \mathbf{X}_i^c) + \lambda tr(\mathbf{W^{*T}}\mathbf{D}^{-1}\mathbf{W^*}) \leq \\ \sum_i L(Y_i, \hat{Y}_i \sum_{c=1}^{C} M_c(\mathbf{W_c}, \mathbf{X}_i^c) + \lambda tr(\mathbf{W}^T\mathbf{D}^{-1}\mathbf{W}).\end{aligned} \tag{3.17}$$

By expanding $tr(\mathbf{W^{*T}}\mathbf{D}^{-1}\mathbf{W^*})$ and $tr(\mathbf{W}^T\mathbf{D}^{-1}\mathbf{W})$ in (3.17), we obtain

$$\begin{aligned}\sum_i L(Y_i, \hat{Y}_i \sum_{c=1}^{C} M_c(\mathbf{W_c^*}, \mathbf{X}_i^c) + \lambda \sum_{g=1}^{G} \frac{\|\mathbf{W^*}_g\|_2^2}{\frac{2}{p}\|\mathbf{W}_g\|_2^{2-p}} \leq \\ \sum_i L(Y_i, \hat{Y}_i \sum_{c=1}^{C} M_c(\mathbf{W_c}, \mathbf{X}_i^c) + \lambda \sum_{g=1}^{G} \frac{\|\mathbf{W}_g\|_2^2}{\frac{2}{p}\|\mathbf{W}_g\|_2^{2-p}}.\end{aligned} \tag{3.18}$$

By adding (3.13) to (3.18), we obtain (3.19) showing the error convergence of our algorithm.

$$\begin{aligned}\sum_i L(Y_i, \hat{Y}_i \sum_{c=1}^{C} M_c(\mathbf{W_c^*}, \mathbf{X}_i^c) + \lambda\|\mathbf{W^*}_g\|_{2,p} \leq \\ \sum_i L(Y_i, \hat{Y}_i \sum_{c=1}^{C} M_c(\mathbf{W_c}, \mathbf{X}_i^c) + \lambda\|\mathbf{W}_g\|_{2,p}.\end{aligned} \tag{3.19}$$

Algorithm 2 Weighted SVM learning.

Input: Input data $\mathbf{X} \in \mathbb{R}^{N \times d}$ and labels $\mathbf{Y} \in \{0,1\}^N$. Regularization parameters λ, p

Output: $\mathbf{W} \in \mathbb{R}^d, b \in \mathbb{R}$

1: Initialize $\mathbf{W} \in \mathbb{R}^d$ and b at random;
2: **repeat**
3: Update $\mathbf{D}$
4: $[\mathbf{W}, b] \leftarrow$ L-BFGS$(E, \frac{\partial E}{\partial \mathbf{W}}, \frac{\partial E}{\partial b})$;
5: **until** Convergence

□

3.4 How to apply the framework: WSVM instantiation

This section introduces the weighting SVM model (WSVM), a linear SVM with group sparsity constraints. WSVM definition shows how to derive a complete model from the general definition ((3.12)).

3.4.1 Model

Linear SVM has demonstrated encouraging results in the context of multimedia classification while limiting the training complexity to $O(n)$ [210]. We consider a linear model to capture the different context information:

$$\forall c \; M_c(\mathbf{W}_c, \mathbf{X}) = \mathbf{X}\mathbf{W}_c. \tag{3.20}$$

In this case, we have $\sum_c M_c(\mathbf{W}_c, \mathbf{X}) = \sum_c \mathbf{X}\mathbf{W}_c = \mathbf{X}\mathbf{W}$. Considering a square hinge loss, our model becomes

$$O(\mathbf{W}, \mathbf{D}) = \sum_i \max(0, 1 - \mathbf{Y}_i(\mathbf{X}_i\mathbf{W} + b))^2 + \lambda tr(\mathbf{W}^T\mathbf{D}^{-1}\mathbf{W}), \tag{3.21}$$

where b is the bias term associated with our global model.

Given algorithm 1, we only need to specify the optimization of (3.21) according

to $(\mathbf{W}, b)$ to obtain an optimization methods of our model. Using a square hinge losss, (3.21) is a convex and smooth optimization problem when $\mathbf{D}$ is fixed. We adopt a direct gradient descent. Such approaches applied of the primal SVM formulation has demonstrated good performance in large scale learning setting [210]. A Quasi-Newtown LBFGS algorithm is used in this work. Compared to a classic SVM, we only need to change the definition of the derivative $\frac{\partial E}{\partial \mathbf{W}}$ to include the sparsity constraints.

$$\frac{\partial E}{\partial \mathbf{W}} = 2\sum_i (\mathbf{X}_i\mathbf{W} + b - \mathbf{Y}_i)\mathbf{X}_i + 2\lambda \mathbf{D}^{-1}\mathbf{W}, \tag{3.22}$$

$$\frac{\partial E}{\partial \mathbf{b}} = 2\sum_i (\mathbf{X}_i\mathbf{W} + b - \mathbf{Y}_i). \tag{3.23}$$

It leads to the definition the WSVM optimization algorithm 2.

3.4.2 A First Application

In this section we introduce a first experiment to demonstrate the capabilities of the WSVM model where we benefit from WSVM to combine multiple space-time contexts on the HMDB dataset (see 2.3.4).

Local dense trajectory features have recently achieved state-of-the-art performance for human action recognition [197]. We therefore choose to model a video as a bag-of local trajectory words. More specifically, we rely on LLC coding and max-pooling to transform local features into a global representation since LLC and max-pooling have demonstrated good performances when they are combined with a linear model [114]. Following [197], we use a vocabulary of size 4000 to compute our signature.

To capture space-time information, we leverage spatio-temporal grids [102]. We apply predefined segmentation grids to the video volume and compute one bag-of-words context per grid cell. We consider 3 different grids, a 1x1x1 segmentation grid, leading to a traditional bag-of-words invariant to space-time transformation but discarding the space-time information, and, a 2x2x2 and 3x3x3 which divide each video axis (x, y and time-dimension) in 2 or 3 cells respectively.

Context	1x1x1 (BoW)	2x2x2	3x3x3
Accuracy	41.6	**44.3**	44.0

Table 3.2: Evaluation of the different spatial context HMDB dataset.

Table 3.2 reports the average accuracies of a SVM model applied on the different contexts. In practice, we use a WSVM with the parameter $p = 2$ (in this case the WSVM is equivalent to traditional SVM) and $\lambda = 0.1$. Table 3.2 shows that 2x2x2 and 3x3x3 spatio-temporal grids context outperforms the bag-of-words on average. But, as Figure 3-4 highlights, space-time grid contexts do not always obtain the best on the individual concepts. Indeed, a BoW signature achieves the best performance for 10 concepts, 2x2x2 signature obtains best accuracies for 22 concepts while 3x3x3 outperforms the two other context on 19 concepts. Space-time grid and BoW therefore appear to be complementary.

p	2	1.5	1	0.5	0.1	0.001
Accuracy	45.1	46.5	**47.4**	46.0	44.3	42.1.

(a) Impact of the parameter p for $\lambda = 0.1$.

λ	0.001	0.01	0.1	1	10
Accuracy	44.4	45.7	**47.4**	46.4	45.8

(b) Impact of the parameter λ for $p = 1$.

Table 3.3: WSVM evaluation.

Table 3.3 investigates the use of a WSVM model for the context combination. Table 3.3a studies the impact of the sparsity parameter p while table 3.3b investigates the cost weighting parameter λ. First, we observe that the different contexts are indeed complementary. A gain of 7% percent, compared to the 2x2x2 grids obtained by considering the different context jointly in the WSVM model, (from 44.3 to 47.4) . In addition, Figure 3.3a shows that by constraining the sparsity in our action model, we can further extend the performance. A WSVM using a $\|.\|_{2,1}$ (*i.e.* $p = 1$) reaches the average accuracy of 47.4 which correspond to a gain of 5% compared to a traditional SVM model using a $\|.\|_2$ norm (equivalent to $p = 2$ with a WSVM). Adding sparsity in the model regularization does improve the performance gain.

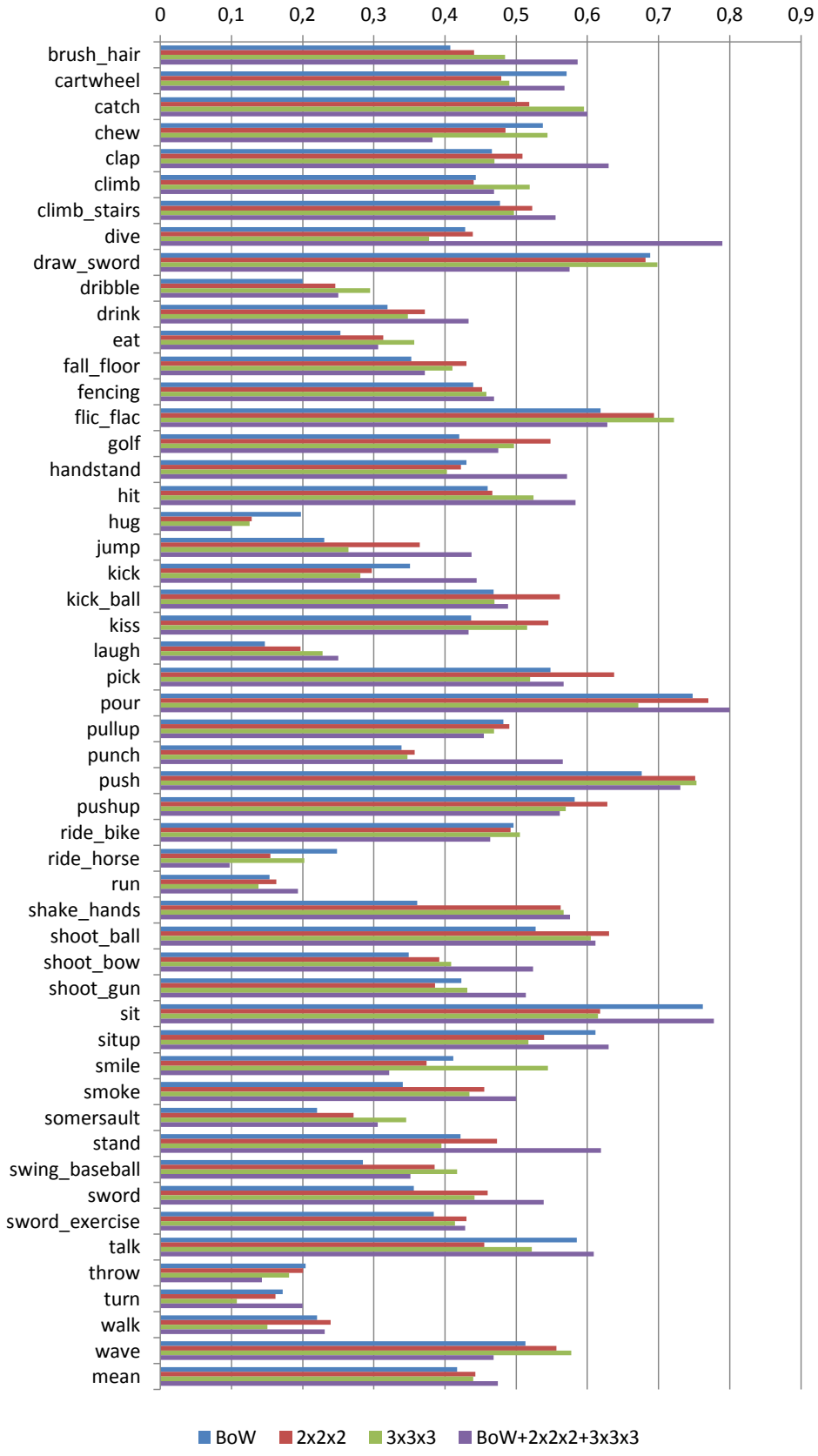


Figure 3-4: Per class average accuracy on the HMDB datasets.

3.5 Conclusion

In this chapter, we introduced a new automated concept annotation framework that leverages various multimedia contexts. Relying on a generalized sparsity regularizer, it automatically learns the optimal context signatures associated with a concept. We also presented a concept annotation formalism. Within this formalism, a context c is determined by two entities, a context extractor function f_c and a model M_c. f_c computes a fixed length vector $X \in \mathbb{R}^{1\times D_c}$ from a video V to identify and summarize the characteristics of a context. M_c is a context model which captures the correlation between the context signatures and the high-level representations, *i.e.* the concepts.

We show in a first experimentation that context selection through group sparsity does help for action recognition in unconstrained videos. A gain of 7% is achieved by a sparse classification model to choose between various fix-grid based representations.

The main challenges of the following chapters will be to identify bottlenecks in the existing multimedia signatures and define new context by specifying f_c and M_c accordingly to improve the multimedia representation. Given the definition of those new contexts, we will experimentally verify the relevance of our model for automated multimedia framework.

Chapter 4

Feature Covariance Context

This chapter proposes a novel video context that focuses on the visual feature inter-dependencies. While existing video signatures [102, 196] model different aspect of the visual content using several descriptors (appearance, motion, acceleration...), they generally don't consider the inter-descriptors linear dependency. Differently, we aim at determining if the descriptor covariance contains discriminative information, useful for automated annotation. To answer this question, we propose in this chapter:

- a novel low-level context capturing the feature descriptor inter-dependency information through covariance;
- a bi-linear learning model which leverages matrix structures.

We evaluate our approach on action recognition datasets and show that considering the covariance information can lead to a gain up to 22%. Covariance information is therefore critical for action recognition.

We start by introducing the motivations behind the features inter-dependency modeling.

4.1 Motivation: Improving the Representation Discriminative Capability

Actions lie at the core of the video concept annotation problem. As Chapter 1 highlights, they are at the center of concept annotation task. By definition, a video is a stream of moving visual images. Being continuous in time, it transcribes the different gestures taking place during the recording. Actions are composed by a set of gestures, and are generally the video very subject. In this chapter, we focus on modeling the video low-level visual signal to characterize action in video.

Definition 5. *Low-level visual signal: values of the different voxels (3D pixels) composing a video.*

Definition 6. *Low-level context: transformation of the low-level signal representation into a new one that preserves significant information while discarding irrelevant detail, to determine what falls in which category.*

Our aim is to design a low-level context which represents actions occurring in videos. We consider the problem of action recognition rather than action detection in order to focus our efforts on the representation, *i.e.* we assume that coarse time delimitation of actions is known.

Although, our proposal can easily be adapted to the action detection problem, where the goal is to recognize and localize actions in videos. One could use sliding temporal windows or more elaborated techniques [219].

4.1.1 Action Representation

An action is defined as a set of movements and postures corresponding to a certain activity. More specifically, in a video, an action is characterized by a combination of local space-time regions with specific appearances and motions. For instance, the action *running* can be coarsely decomposed in three space-time regions characterizing

Running Kick Ball

Figure 4-1: Decomposition of an action into spatio-temporal regions. Red rectangles identify the main action regions while red arrows correspond to the region principal motion.

the legs and the torso of the human body as Figure 4-1 illustrates. Similarly, the action *kick ball* is divided into two regions: the human legs and the soccer ball.

Hence, to represent an action, we need to describe the content of local space-time regions composing the action. Only the joint analysis of video appearance and motion, which captures posture and movement information, allows the discrimination of actions. Action such as *running* and *walking* may share similar visual appearance, they have different motion patterns. On the other hand, actions like *kick* and *kick ball* may share common motion, the visual presence of a soccer ball allows to differentiate them.

4.1.2 Bag-of-Words: First-Order Statistics

Many researchers [9, 37, 102, 126, 129, 130, 179, 197, 205] have investigated low-level signal representations that capture motion and appearance information. Due to its robustness to unsemantic variation (illumination change, clutter, occlusion...), Bag-of-Words (BoW) [170] has been adopted as the dominant paradigm for video representation.

A BoW is computed in 3 steps: (1) local feature extraction, (2) local feature coding and (3) local feature pooling. Let $\mathbf{D} = \{\mathbf{d}_i\}_{i\in[1,M]}$, with $\mathbf{d}_i \in \mathbb{R}^{1\times D}$, be a bag of descriptors characterizing local features. Equations (4.1) and (4.2) formalize the

computation of a BoW signature $\mathbf{X}$ with average pooling and the max pooling :

$$\mathbf{X} = \frac{1}{M}\sum_{i=1}^{M}\mathbf{c}_i \tag{4.1}$$

$$\mathbf{X} = \max_{i\in[1,M]}\mathbf{c}_i. \tag{4.2}$$

In (4.2), $\mathbf{c}_i \in \mathbb{R}^{1\times K}$ is a code associated with the descriptor, *i.e.* $\mathbf{d}_i$: $\mathbf{c}_i = code(\mathbf{d}_i)$ where function $code : \mathbf{R}^{1\times D} \rightarrow \mathbb{R}^{1\times K}$ is any coding scheme such as hard-Coding, sparse coding or LLC coding [63, 170, 200, 210, 217]. Average pooling and max-pooling compute, respectively, the mean and the maximum over the coded descriptors. They capture first-oder statistics of feature codes.

Statement: BoW representation does not model explicitly the local descriptor covariances. A video is represented using only the first order statistics of coded features, and, traditional coding schemes don't consider explicitly the local descriptors covariation [170]. Originally BoW encodes local features into a set of code $\mathbf{C} = \{\mathbf{c}_i\}_{i=1}^{M}$ using a vocabulary $\mathbf{V} = \{\mathbf{v}_j\}_{j=1}^{K}$, with $\mathbf{v}_j \in \mathbb{R}^{1\times D}$ and $\mathbf{c}_i \in \mathbb{R}^{1\times K}$. The vocabulary is constructed by minimizing the descriptor cumulative reconstruction error:

$$\{\hat{\mathbf{V}}, \hat{\mathbf{C}}\} = \arg\min_{\mathbf{V},\,\mathbf{C}} \sum_{i=1}^{M} \|\mathbf{d}_i - \mathbf{c}_i\mathbf{V}\|_2^2, \tag{4.3}$$

(4.3) is solved using a a k-mean clustering algorithm. Here, k-means can be seen as a classic Gaussian Mixture Model which constrains the mixture covariances to be identity matrices. Hence, it does not take into account the descriptor covariances to build the visual vocabulary.

The k-mean clustering results in a set of k Voronoi cells that segment the descriptor space [118]. Each visual word $\mathbf{v}_j$ is the mean of the elements falling in one Voronoi cell. Given a vocabulary $\mathbf{V}$, BoW hard-coding scheme [170] assigns a feature $\mathbf{d}_i$ to

its nearest word present in the codebook:,

$$\mathbf{c}_i \in \{0,1\}^K \text{ with } \mathbf{c}_{i,j} = \begin{cases} 1 & \text{if } j = \arg\min_{[1,K]} ||\mathbf{d}_i - \mathbf{v}_j||_2^2 \\ 0 & \text{otherwise.} \end{cases} \tag{4.4}$$

Since a visual word $\mathbf{v}_j$ represents only the Voronoi cell mean, hard-coding discards the descriptor covariance information. Other coding schemes such as sparse coding (SP) [210], or locality-constrained coding (LLC) [63, 200, 217] has been proposed by the community. Those schemes encode a feature using a sparse combination of visual words. However, they still discard the covariances information the vocabulary computation or features coding. Consequently, BoW doesn't model explicitly the feature covariance information.

4.1.3 Covariance: Higher-Order Statistics

BoW representation misses the higher-order statistical information such as covariance. We state that covariance statistical information provides key discriminative information for action characterization, thus, BoW lacks of covariance modeling weaken its discriminative power.

For instance, let's consider a standard Histogram of Gradient (HoG) and Histogram of Flow (HoF) descriptors. HoG and HoF are respectively local gradient orientation histogram and local flow motion histogram. Covariance captures the linear dependencies between the HoG and HoF dimensions. Contrary to HoG or HoF which count the number of time a low-level edge or motion pattern occurs in a local patch, it captures mid-level pattern that characterizes jointly the motion and appearance information. Since covariance characterizes higher-order information, it is likely to be more discriminative.

To illustrate the discriminative power of covariance statistics, we propose a simple experimentation which is illustrated in Figure 4-2. We consider three action classes, *Run*, *Kick Ball* and *Walk*, out of the HMDB dataset [95] (see section 2.3). For each

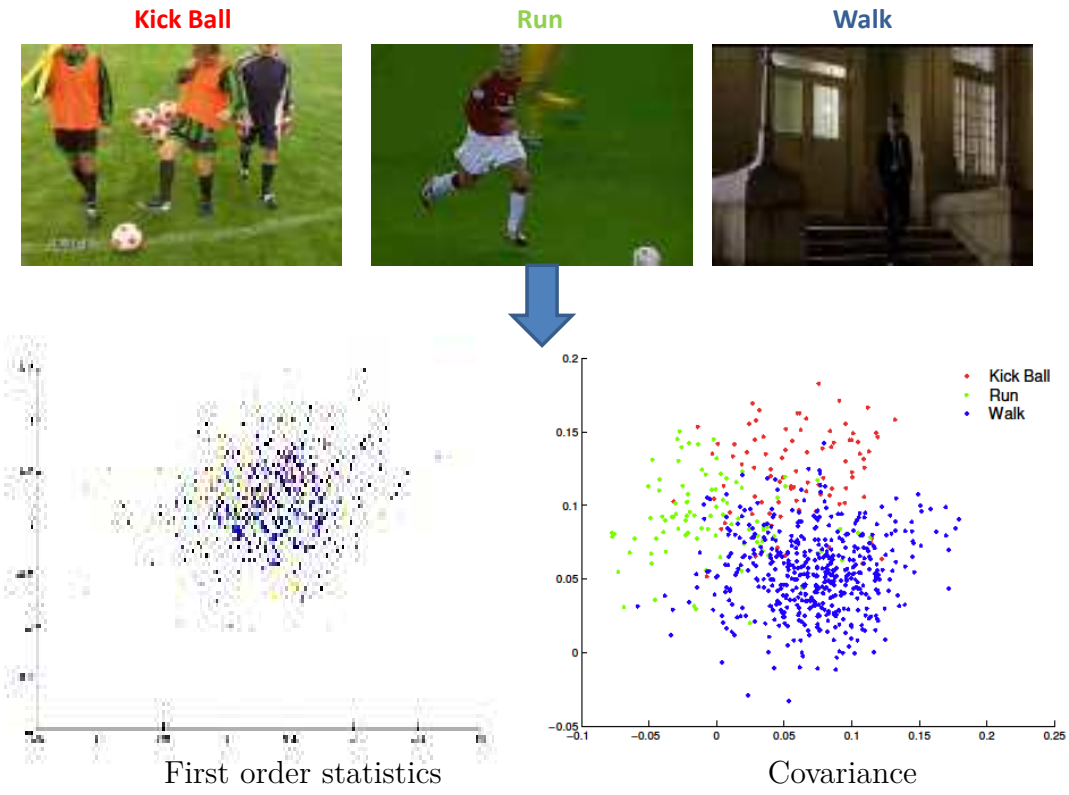


Figure 4-2: Illustration of the covariance discriminative capacity. We consider an action recognitions problem with three classes. We extract the Histogram of Gradient and Histogram of Flow of video local trajectory features. We aggregate the local descriptors per video using simple first order statistic (average) and covariance. We apply a Linear Discriminant Analysis on both aggregation methods. This figure shows that the separation between the different classes is more apparent within the covariance representation.

video, we extract the HoG and HoF descriptors associated with dense trajectories [197] and aggregate them into fixed length signature using a simple first order statistic (average) or their covariances. We project both signatures in the 2D space using linear discriminative analysis (LDA). It is noticeable that the covariance signatures are more linearly separable than the first order statistics. This example tends to show that the covariance between multiple descriptor can be useful for action classification.

4.1.4 Our Contributions:

We aim at extracting a context signature that represents an action from the low-level visual signal. In addition to first order statistics, we explore higher-order information,

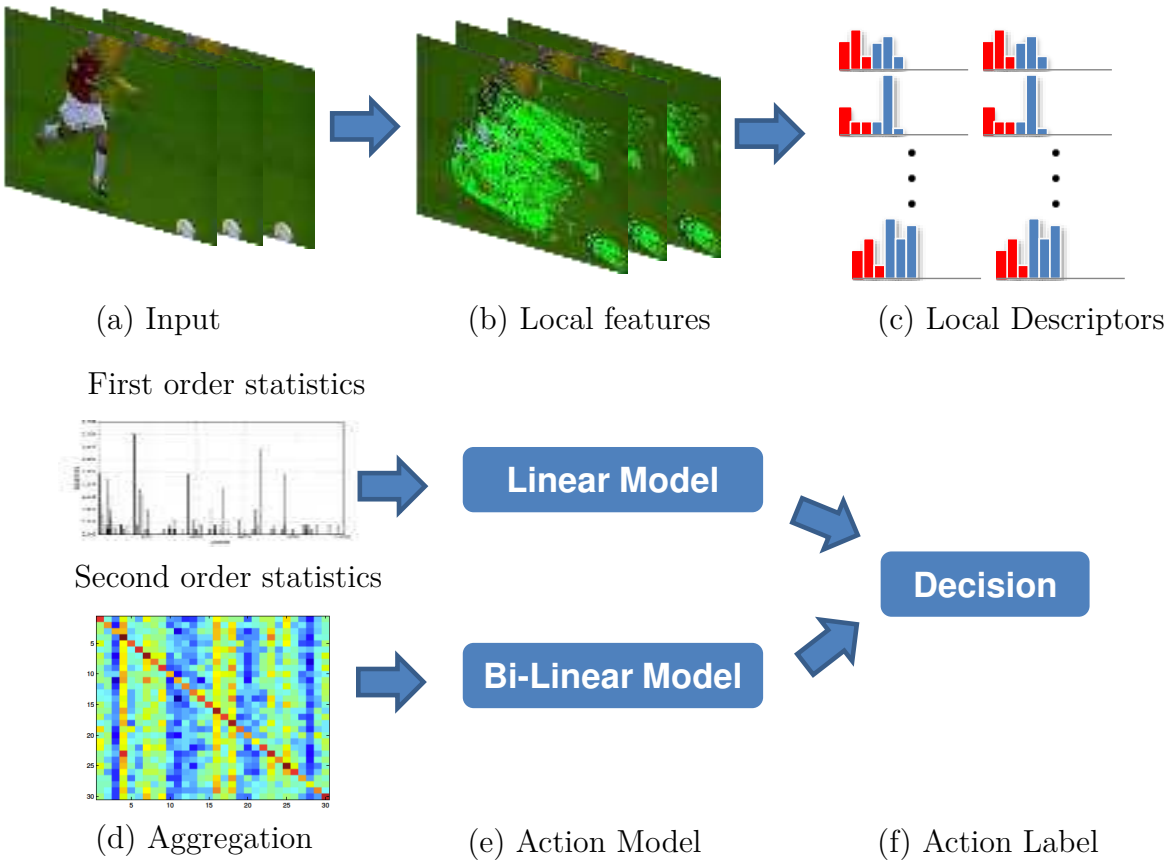


Figure 4-3: Synopsis of the covariance and BoW combination.

Figure 4-3 summarizes our approach:

- we introduce a covariance context capturing the linear dependencies of the different video local descriptors;
- we model the correspondence between action and estimated covariance matrices using a bi-linear model to leverage the 2D data structure;
- We combine the covariance context and the first-order-representation using the multi-context framework developed in Chapter 3.

4.2 Related Work

Several works have investigated the use of covariance for annotation in still images [27, 84, 169, 187, 201, 218], this section thus proposes a critical review of these previous approaches. We also examine the state-of-art of bi-linear model [117, 148, 182, 206].

The differences of our approach with state-of-art methods is synthesized in Table 4.1.

4.2.1 Covariance Representation

Covariance information has originally been considered to capture the spatial correlation between local patches in image representations [84, 218]. Karling [84] learned a linear filter bank that models the variance of filter responses to characterize spatial correlation. Yu [218] proposed a two-layer sparse coding framework for image classification. Our goal significantly differs with those previous works. We want to describe the inter-dependencies between different local feature descriptors, not their spatial context.

Covariance has also been used to characterize local region in images [27, 169, 187, 201]. Tuzel [187] introduced a covariance descriptor which captures the linear-dependencies between image pixels using their intensity, gradient and RGB values. Sivalingam [169] proposed to code region descriptors leveraging a sparse vocabulary to increase their robustness. In a same way, Wang [201] uses a generative model to learn the region covariance pattern. Closest to our approach, Authors of [27] have investigated the use of covariance with local features rather than pixel-wise feature to characterize free-form segmented regions in images.

Three main differences exist with our approach. First, all the previous works, except the proposition of Carreira [27], constructs region covariance using the pixel-values directly. Differently, we rely on local-features that characterize patches in a video to select more robust and discriminative information. Moreover, those works are designed to characterize specific region in image while we aim at describing the global video content. Finally, they are all using classic classification algorithm (Nearest-Neighbor, SVM...) to annotate images. Such algorithms rely on vector features, and therefore, discard the covariance matrix structure. Differently, we take advantage of a bi-linear maximum margin model preserving the 2D structure covariance matrix.

Fisher Vector [146] is an alternative to the BoW representation which models a vi-

sual vocabulary using a Gaussian Mixture Model (GMM) and attributes a descriptor to a visual word using mean and variance. Because standard Fisher Vector assumes a GMM with diagonal matrices, statistical information related to correlation is not modeled. Of course, in theory, the Fisher vector could benefit from such information by modeling a GMM with full covariance matrices; however, the cost would be prohibitive for real world problem.

Regarding video, Guo *et al.* [54] have introduced a low-level representation using covariance. However, they rely on holistic silhouette feature to compute their covariance representation. Due to this holistic aspect, their approach is limited to constrained videos. Differently, we leverage local descriptor to handle unconstrained videos.

	Signature	Type	Correlation	Application	Classifier
Karling [84]	Global	Hand-Crafted	Spatial	Image	None
Yu [218]	Global	Hand-Crafted	Spatial	Image	SVM
Tuzel [187]	Region	Hand-Crafted	Pixel	Image	NN
Sivalingan [169]	Region	Hand-Crafted	Pixel	Image	NN
Wang [201]	Region	Learned	Pixel	Image	Vote
Carreira [27]	Region	Hand-Crafted	Local Features	Image	SVM
Guo [54]	Global	Hand-Crafted	Holistic	Video	Sparse-NN
Our Approach	Global	Hand-Crafted	Local Features	Video	Bi-linear

Table 4.1: Summary of other approaches relying on feature covariance. SVM stands for Support Vector Machine, NN stands for Nearest Neighbors.

4.2.2 Bi-linear model

Covariance representation leads to 2D matrix features. However, linear model do not consider the 2D aspect of the features [117]. Bi-linear model have been proposed to preserve the correlation within matrix data structure. They have been originally introduced to the vision community by Tenenbaum *et al.* [182] to model data gen-

erated from multiple linear factors in the context of density estimation. Relying on the bi-linear model, the work of Wolf *et al.* [206] demonstrated the usefulness of matrix representation in visual images classification. More recently, Ma *et al.* [117] proposed a bi-linear classification model for images regression in a semi-supervised setting. One novelty of their approach is the use of compound regression to increase the model degree of freedom. Closest to our work, Pirshiavash *et al.* [148] introduce a discriminative bi-linear approach for classification. A bi-linear SVM classifier is able to classify 2D matrices data such as images, or in our case, covariance matrices. The approach is encouraging, however, as its training process depends on traditional SVM solver designed for vector representation, Pirshiavash bi-linear-SVM learning therefore still needs to transform matrices into vectors at some point.

By contrast, we propose a new bi-linear SVM learning optimization which doesn't requires any vectorization. It drastically reduces the classifier complexity. In addition, we adapt the multiple-compounds to our bi-linear model to increase its expressivity. We also provide a theoretical reason that explains the gain of performance obtained by multiple-compounds model.

4.3 Covariance Context

In this section we introduce a context extractor function $f : \mathbb{V} \rightarrow \mathbb{R}^{1\times(D\times D)}$ which specifies the local descriptor covariance information from a video.

We consider a set $\mathbf{D} = \{\mathbf{d}_i\}_{i\in[1,M]}$ computed from a video, where $\mathbf{d}_i \in \mathbb{R}^{1,D}$. Each $\mathbf{d}_i$ is the contenation of P local descriptors, $\mathbf{d}_i = [\mathbf{d}_i^p]_{p\in[1,P]}$. A subvector $\mathbf{d}_i^p$ is a local descriptor capturing one particular aspect of a local video subvolume (appearance, motion, acceleration, position...).

To highlight the relation existing between the different descriptors, we compute

their covariations. We introduce the function $cov : \mathbb{R}^D \rightarrow \mathbb{R}^{1\times(D\times D)}$ such that:

$$cov(d_i) = (\mathbf{d}_i - \mu)^t(\mathbf{d}_i - \mu), \tag{4.5}$$

with μ being the average descriptor of $\mathbf{D}$:

$$\mu = \sum_{j=1}^{M} \mathbf{d}_j. \tag{4.6}$$

cov characterizes the linear dependencies between the different dimension of $\mathbf{d}_i$. It is a measure of how much two $\mathbf{d}_i$ dimensions evolve together. The covariance is positive if the greater values of one dimension correspond with the greater values of another dimension, *i.e.* the descriptor dimensions show similar behavior. On the other hand, when the greater values of one dimension mainly correspond to the smaller values of the other, *i.e.* the dimensions demonstrate opposite behavior, their covariance is negative. The sign of the covariance therefore shows the tendency in the linear relationship between the descriptor dimensions.

After the descriptor covariance computation, a video is represented by a set of covariance matrices $cov(\mathbf{D}) = \{cov(\mathbf{d}_i)\}_{i\in[1,M]}$. This representation is unsuitable for classification for two reasons. First, there are a variable number of local features extracted from videos, M is video dependent. This length variability prevents from using many traditional classifiers such as SVM for annotation directly on $cov(\mathbf{D})$. In addition, the set of covariance matrices dimension is $M \times D^2$ which is likely to be very large. To tackle those issues, we apply a pooling operation to transform set of covariance matrices into fixed length signature having a lower dimension. Inspired by traditional bag-of-words, we consider average and max pooling:

$$\mathbf{X} = \frac{1}{M}\sum_{i=1}^{M} cov(\mathbf{d}_i), \tag{4.7}$$

$$\mathbf{X} = \max_{i=1}^{M} cov(\mathbf{d}_i). \tag{4.8}$$

When average pooling is used with the descriptor covariances, the context extractor

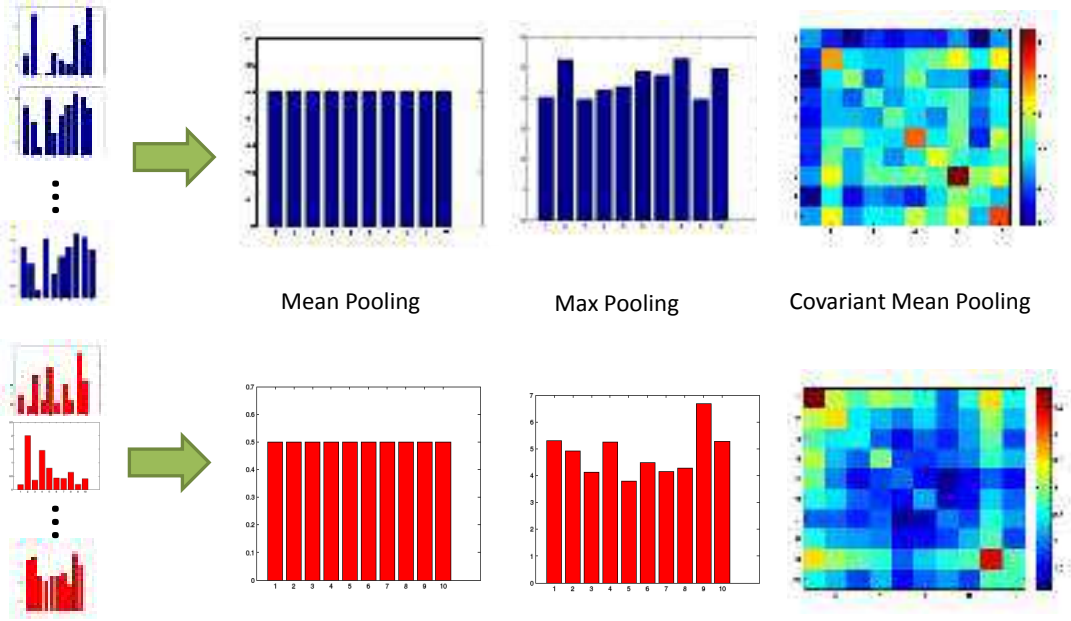


Figure 4-4: Mean, max and average covariance pooling applied to two synthetic sets of local descriptors. Descriptors are distributed accordingly to multi-dimensional Gaussian having the same mean but different covariances. While mean and max pooling don't exhibit strong differences between the two distributions, covariance pooling is able to capture the distribution specificities.

f computes the Sample Covariance Matrix (SCM) estimator [174].

In the following, we denote by covariance pooling the application of max or average pooling to the feature covariance matrices.

Differently to the Bag-of-words model, we do not encode the local features prior to the pooling. Indeed, the goal of coding is to prune irrelevant details from local features while keeping discriminative information. To enhance the features discriminative power, coding applies a non-linear operator that projects local features in a larger space. By construction, covariance pooling already applies a non-linear operator which project the local features in a larger space to capture the descriptor higher order statistics. It limits the need of coding in this case.

Pooling extracts statistics over a set of local descriptors. Since our goal is to discriminate between several actions, the action-conditional statistics extracted should be different. To illustrate the effect of the covariance pooling operation, we consider

two different synthetic set of descriptors in Figure 4-4. The two sets follow multivariate Gaussian distributions sharing similar mean but having different covariances. Figure 4-4 compares the average covariance pooling with simple max and average pooling applied on the descriptor directly. It shows that average pooling is not able to discriminate between the two distributions. While having small differences, max-pooling still lead to similar representation. Covariance pooling, differently, shows strong differences between the two distribution representations. By focusing directly on the covariance representation, we expect to increase the action separability.

4.4 Covariance Model

We propose a classifier that learns the correspondence between the pooled covariance matrices and the actions. We consider a set of covariance signatures $\mathbf{X} \in \mathbb{R}^{N\times(D\times D)}$ and their corresponding labels $\mathbf{Y} = \{\mathbf{Y}_i\}_{i\in[1,N]}$. $\mathbf{X_i} \in \mathbb{R}^{1\times(D\times D)}$ is a sample covariance matrix estimated through max or average pooling from one video. We abuse the notation so $\mathbf{X_i}$ designs the covariance signature in $\mathbb{R}^{D\times D}$ space directly. Our goal is to learn the parameter set associated with our classifier model.

To characterize the correspondence between action and covariance signature, we introduce a maximum margin bi-linear model that exploits the 2D structure of covariance matrices, and, propose a new optimization algorithm to learn the bi-linear SVM parameters. In addition, we adapt the multiple-compound aspect introduced by Ma *et al.* [117] to maximum margin loss function in order to increase to the expressiveness of our model.

4.4.1 Limitation of Linear Model for Covariance Matrices

Most of the traditional classifiers, such as linear SVM [2, 210] require 1-dimensional vector. To apply such models, one needs to "flatten" the correlation matrices into vectors prior to the classification. We flat the covariance signatures with a vectorization function, vec : $\mathbb{R}^{D\times D} \rightarrow \mathbb{R}^{D^2}$, which simply concatenates each row (or column)

of a matrix. We denote by $\mathbf{X}^v \in \mathbb{R}^{N \times D^2}$ the vectorized signatures, *i.e.* $\mathbf{X}_i^v = \text{vec}(\mathbf{X}_i)$.

Vectorization has some limitations. Firstly, the 2D spatial organization between the matrix coefficients is broken. In addition, because of the row or column concatenation, vectorization implies a quadratic augmentation of the signature dimension, heavily increasing the model complexity. A large model complexity can lead to overfitting and impact the performance.

The complexity increase caused by the signature dimensionality augmentation can be quantified. The classifier complexity corresponds to the maximum number of samples that can be exactly classified given any possible label assignments. This capacity is measured by the VC dimension. We consider a classification model M and its parameter vector $\mathbf{W}$. $\mathbf{W}$ is said to shatter a signature set $\mathbf{X}^v$ if for all the possible label combinations $\mathbf{Y}$, a parameter vector $\mathbf{W}$ exists such that

$$\forall i \in [1, N],\ M(\mathbf{X}_i^v, \mathbf{W}) = Y_i. \tag{4.9}$$

The complexity of a model, or VC dimension is the maximum number of samples N such as $\mathbf{X}^v$ is shattered by the model M . It has been shown that the VC dimension of linear model with bias that classifies D-dimensional signature is $D+1$ [206]. Thus, in the case of vectorized signature, we obtain a linear SVM classifier having a VC dimension of $D^2 + 1$.

VC dimension directly relates to the classifier performance. Most naturally, one can estimates the classifier performance by computing expected risk R_{test}. R_{test} is the misclassification rate on the testing examples. However, since the complete set of testing examples is unknown, R_{test} is not tractable in practice. Fortunately, R_{test} is bounded by the empirical training error and the model complexity [25, 192]:

$$R_{\text{test}}(\mathbf{W}) \leq R_{emp}(\mathbf{W}) + \sqrt{\frac{h(log(2N/h) + 1) - log(\eta/4)}{N}}. \tag{4.10}$$

Here, $R_{\text{test}}(\mathbf{W})$ is the expected testing error of $\mathbf{W}$, $R_{\text{emp}}(\mathbf{W}) = \sum_{i=1}^{N} L(Y_i, M(\mathbf{W}, \mathbf{X}_i^v))$

is the empirical training error, h is the Vapnik–Chervonenkis dimension (VC dimension) characterizing the model complexity and η a parameter controlling the inequality confidence [192].

(4.10) shows that a classifier must find a good trade-off between the empirical risk minimization and the model complexity. When a model is "simple", the VC dimension being low, it is likely to have a good generalization of the classification performance on unseen data.

Since, vectorization quadratically increase the signature sizes, it also increases the linear VC dimension to $D^2 + 1$. If the original signature size is too large, it results in a complex classifier model prone to overfitting.

4.4.2 Multi-Compound Bi-Linear Model

Bi-linear model have been proposed to preserve the correlation within the matrix data structure while significantly reducing the classifier VC dimension.

Model

We consider two classification vectors, $\mathbf{u} \in \mathbb{R}^{1\times D}$ and $\mathbf{v} \in \mathbb{R}^{1\times D}$. Following Pirshiavash *et al.* [148], we propose the following bi-linear model:

$$M(\mathbf{u}, \mathbf{v}, \mathbf{X}_i) = \mathbf{u}\mathbf{X}_i\mathbf{v}^T, \tag{4.11}$$

which solves the following functional:

$$\{\hat{\mathbf{u}}, \hat{\mathbf{v}}, \hat{b}\} = \arg\min_{\mathbf{u},\mathbf{v},b} \sum_{i=1}^{N} L(Y_i, \mathbf{u}\mathbf{X}_i\mathbf{v}^T + b) + \lambda\|\mathbf{u}^T\mathbf{v}\|_F^2, \tag{4.12}$$

b is the model bias term and $\|.\|_F$ is the Frobenious norm which is the equivalent of ℓ_2 norm for matrix:

$$\|\mathbf{W}\|_F = \sqrt{\sum_i \sum_j \mathbf{W}_{i,j}^2}. \tag{4.13}$$

Since we are interested in a maximum-margin model, we consider the square hinge-loss as L in (4.12). However, any convex and smooth loss functions can be used.

We observe in (4.13) that each coefficient in the vector $\mathbf{u}$ (respectively $\mathbf{v}$) is applied on one line (respectively column) of the covariance matrix $\mathbf{X}_i$. By contrast each coefficient of a linear vector model corresponds solely to one matrix coefficient. Indeed for a linear model $\mathbf{W}$, we have:

$$M(\mathbf{W}, \mathbf{X}_i^v) = \mathbf{X}_i^v \mathbf{W}, \tag{4.14}$$

where, $\mathbf{X}_i^v$ is the vectorized covariance matrix feature. In this sense, bi-linear SVM leverages the matrix structure contrary to the linear model.

Bi-linear model reduces considerably the model complexity compared to the vectorized linear SVM. While the linear model VC dimension of a vectorized $\mathbf{R}^{D\times D}$ matrix is $D^2 + 1$, it has been shown that the bi-linear SVM model VC dimension is only $2D$ [206]. Such a strong reduction in the VC dimension can lead to classifiers which are too simple to characterize the training dataset. Indeed, as the bound on the expected risk shows (4.10), the classifier performance is a trade-off between the empirical risk and the model complexity. Despite having a low VC dimension, if a model becomes too simple the empirical risk can suffer and degrades the classifier overall performance.

Ideally, we would like to design a classifier having the lowest VC dimension while still fitting the training dataset. To have finer control of its complexity, we introduce multiple compounds in our model . We consider two groups of classification vectors $\mathbf{U} = \{\mathbf{u}_i\}_{i=1}^C$ and $\mathbf{V} = \{\mathbf{v}_i\}_{i=1}^C$ such that $\mathbf{u}_i \in \mathbb{R}^{1\times D}$ and $\mathbf{v}_i \in \mathbb{R}^{1\times D}$. Using the multiple compounds vectors, our objective functional becomes:

$$\{\hat{\mathbf{U}}, \hat{\mathbf{V}}, \hat{b}\} = \min_{\mathbf{U},\mathbf{V},b} \sum_{i=1}^{N} L(Y_i, \sum_{c=1}^{C} \mathbf{u}_c \mathbf{X}_i \mathbf{v}_c^T + b) + \lambda \sum_{c=1}^{C} \|\mathbf{u}_c^T \mathbf{v}_c\|_F^2. \tag{4.15}$$

Compared with single classification model $M = 1$, multiple compounds model (4.15)

provides a larger search space for the solution. We can immediately see that the VC dimension of (4.15) becomes $2CD$ where C is the number of compound components. With (4.15), we are able to choose the complexity of our model given the classification task.

Optimization

To learn the bi-linear model parameter, we need to solve the following equation:

$$\{\hat{\mathbf{U}}, \hat{\mathbf{V}}, \hat{b}\} = \min_{\mathbf{U},\mathbf{V},b} \sum_{i=1}^{N} L(Y_i, \sum_{c=1}^{C} \mathbf{u}_c \mathbf{X}_i \mathbf{v}_c^T + b) + \lambda \sum_{c=1}^{C} \|\mathbf{u}_c^T \mathbf{v}_c\|_F^2. \tag{4.16}$$

(4.16) defines a bi-convex problem, *i.e.* (4.16) is convex when $\mathbf{U}$ or $\mathbf{V}$ is fixed. Bi-convex problem has been well-studied in the optimization literature [53]. While not convex, such problems admit efficient coordinate descent algorithms that solve a convex program at each step [148]. We therefore alternatively relax (4.16) by fixing one of the model variables. We perform an iterative coordinate descent (cf algorithm 3). We start by relaxing the problem by fixing $\mathbf{V}$ (which is initialized at random) and solve (4.16) according to $\mathbf{U}$ and b. Let's introduce $\mathbf{u} = [\mathbf{u}_c]_{c\in[1,C]}$ (repectively $\mathbf{v} = [\mathbf{v}_c]_{c\in[1,C]}$), the vector concatenating all the $\mathbf{u}_c$ (respectivelly $\mathbf{v}_c$) components. For clarity we define $\mathbf{F}_i \in \mathbf{T}^{DC\times 1}$ such that

$$\mathbf{F}_i = [\mathbf{X}_i \mathbf{v}_c^T]_{c\in[1,C]}. \tag{4.17}$$

(4.16) becomes:

$$\{\hat{\mathbf{U}}, \hat{b}\} = \arg\min_{\mathbf{U},b} \sum_{i=1}^{N} L(Y_i, \mathbf{u}\mathbf{F}_i + b) + \lambda \sum_{c=1}^{C} Tr(\mathbf{u}_c^T \mathbf{v}_c \mathbf{v}_c^T \mathbf{u}_c). \tag{4.18}$$

We then defined the diagonal block matrix $\mathbf{D}$:

$$\mathbf{D} = \begin{pmatrix} \mathbf{v}_1^T \mathbf{v} \mathbf{I}_D & & \\ & \ddots & \\ & & \mathbf{v}_M^T \mathbf{v}_M \mathbf{I}_D \end{pmatrix}, \tag{4.19}$$

where, $\mathbf{I}_D \in \mathbf{R}^{D \times fgD}$ is the identity matrix . We deduce a new regularization term:

$$\sum_{i=1}^{C} \|\mathbf{u}_c \mathbf{v}_c^T\|_F^2 = \sum_{i=1}^{M} Tr(\mathbf{u}_m \mathbf{v}_m^t \mathbf{v}_m \mathbf{u}_m^t) = Tr(\mathbf{u}\mathbf{D}\mathbf{u}^T). \tag{4.20}$$

We replace the regularization term of (4.18) with (4.20) to obtain:

$$\{\hat{\mathbf{U}}, \hat{b}\} = \arg\min_{\mathbf{U},b} \sum_{i=1}^{N} L(Y_i, \mathbf{u}\mathbf{F}^i + b) + \lambda Tr(\mathbf{u}\mathbf{D}\mathbf{u}^T). \tag{4.21}$$

Since $\mathbf{V}$ is fixed in (4.21), $\mathbf{D}$ is also known. As a consequence the minimization of (4.21) according to $\mathbf{U}$ and b is a smooth optimization problem. We actually notice that (4.21) is exactly the same optimization problem than the WSVM model (cf section 3.4). We can therefore solve it using the similar quasi-Newton gradient-descent approach. We now need to minimize (4.16) according to $\mathbf{V}$ with $\mathbf{U}$ is fixed. The method can easily be deduced from the previous one by symmetry.

Algorithm 3 Bi-linear SVM optimization.

Input: Signatures $\mathbf{X} \in \mathbb{R}^{N \times d}$ and labels $\mathbf{Y} \in \{0,1\}^N$. Regularization parameters λ
Output: $\mathbf{u}$, $\mathbf{v}$, b

1: Initialize $\mathbf{u}$, $\mathbf{v}$, b at random;
2: **repeat**
3: $\mathbf{D} = \begin{pmatrix} \mathbf{v}_1^T \mathbf{v} \mathbf{I}_D & & \\ & \ddots & \\ & & \mathbf{v}_M^T \mathbf{v}_M \mathbf{I}_D \end{pmatrix}$ and $\forall i\ \mathbf{F}_i = [\mathbf{X}_i \mathbf{v}_c^T]_{c \in [1,C]}$
4: Perform $\arg\min_{\mathbf{u},b} \sum_{i=1}^{N} L(Y_i, \mathbf{u}\mathbf{F}_i + b) + \lambda tr(\mathbf{u}^T \mathbf{D} \mathbf{u})$;
5: $\mathbf{D} = \begin{pmatrix} \mathbf{u}_1^T \mathbf{u} \mathbf{I}_D & & \\ & \ddots & \\ & & \mathbf{u}_M^T \mathbf{u}_M \mathbf{I}_D \end{pmatrix}$ and $\forall i\ \mathbf{F}_i = [\mathbf{u}_c \mathbf{X}_i]_{c \in [1,C]}$
6: Perform $\arg\min_{\mathbf{v},b} \sum_{i=1}^{N} L(Y_i, \mathbf{F}_i \mathbf{v}^T + b) + \lambda tr(\mathbf{v}^T \mathbf{D} \mathbf{v})$;
7: **until** Convergence

Algorithm 3 alternatively fix $\mathbf{U}$ or $\mathbf{V}$ to solve (4.16). Convergence proof of algorithm (3) is similar to the one provided in [117].

4.4.3 Integration in the Multiple Context Cues Model

The Bi-linear model only captures information related to the covariance. However, we want to leverage both first order and covariance contexts for action annotation. We take advantage of the multiple contexts model defined in Chapter 3 to combine both contexts.

To integrate the bi-linear model in the multiple contexts framework, we defined the low-rank matrix $\mathbf{W}_c = \mathbf{u}_c^T \mathbf{v}_c$. We deduce that (4.15) is equivalent to

$$\{\hat{\mathbf{W}}, \hat{b}\} = \arg\min_{\mathbf{W},b} \sum_{i=1}^{N} L(Y_i, \sum_{c=1}^{C} Tr(\mathbf{W}_c \mathbf{X}_i) + b) + \lambda \sum_{c=1}^{C} \|\mathbf{W}_c\|_F^2 \tag{4.22}$$

Abusing the notation, $\|\mathbf{W}_c\|_2 \sim \|\mathbf{W}_c\|_F$, which is true when $\mathbf{W}_c$ is a vector, we see that we can integrate our model directly in the general context-based classification model.

4.5 Covariance Context Added Value: Evaluation

In this section we evaluate the covariance context and bi-linear SVM performances on action recognition datasets. We start by introducing the different experimental setting.

4.5.1 Implementation Detail

Dense trajectory features have recently shown state-of-the-art performance for human action recognition [197]. They highlight space-time regions keeping a certain visual consistency through time. They are used as building block of our video signature. Keypoints are densely sampled at multiple spatial scales in each of video frames. Dense optical flow using Farneback algorithm [41] is used to match a point from a frame f to the next frame $f + 1$. Trajectories are built by accumulating point correspondences over successive frames. Motion vectors (Track), Histogram of gradient (HoG), histogram of flow (HoF) and motion boundary histogram (MbH) are used as descriptor [197].

To obtain a bag-of-words representation, we take advantage of locality constrained coding (LLC) and max-pooling which have demonstrated encouraging performance when combined with a linear SVM [114]. LLC coding is obtained by restricting the probabilistic soft coding to the 10 nearest words. As specified by Wang [197], we use a codebook of size 4000 to perform the BoW coding, which is a good trade-off between efficiency and performance. Max-pooling is then used to deduce a fixed-length signature.

The covariance matrices domain does not form a vector space (they are not closed under multiplication by negative scalars); they form a Riemannian manifold. Classification problems on covariance manifolds can be converted into vector-space. Following [54], we apply a matrix logarithmic function on the average covariance matrix. The matrix logarithm maps the Riemannian manifold of symmetric non-negative definite matrices to the vector space of symmetric matrices.

We adopt a one-versus-all classification scheme to combine the action binary classifiers. Our approach is evaluated on two standard human action datasets: KTH, and HMDB. Average accuracies are reported for both the datasets.

4.5.2 Does Covariance Information bring discriminative information?

In a first experimentation, we want to determine if covariance signature contains discriminative information useful for action recognition. We compare the max and average covariance signatures with traditional Bag-of-words representations.

For a fair comparison, we use an identical SVM classifier for all three representations. A SVM learns the correlation between the low-level context signatures and the high-level actions. In practice, we use the WSVM formulation with $p = 2$ to obtain a traditional SVM. The SVM regularization parameter λ is set to 0.1 which had empirically demonstrated good performances, see section 3.4.2. We apply a vectorization operation on the covariance matrices prior to the classification.

	BoW	cov-avg	cov-max
KTH	93.7	94.6	**95.4**
HMDB	41.6	44.5	**45.0**

Table 4.2: Average Accuracy for the BoW, cov-avg and cov-max representation.

Table 4.2 reports the averages accuracies for the Bag-of-Word representation (BoW) and the average (cov-avg) and max covariance pooling (cov-max). Table 4.2 shows that covariance achieves competitive performances on both dataset. Indeed, cov-max outperforms a traditional BoW by 1.8% on the KTH dataset, and by 8% on the HMDB dataset. It therefore demonstrates that covariance statistics capture discriminative information in the context of action recognition. Moreover, the fact that covariance outperforms BoW on the challenging HMDB dataset shows that they are robust enough to recognize actions in a realistic setting.

4.5.3 Analysis on a Constrained Dataset

	Boxing	Handclapping	Handwaving	Jogging	Running	Walking
BoW	99.3	95.1	95.8	95.1	78.4	100
cov-avg	100	91.6	93.8	95.8	86.8	100
cov-max	99.3	95.1	93.7	96.5	86.1	100

Table 4.3: Per action average accuracy on the KTH dataset.

Jogging Running

Figure 4-5: Illustration of the KTH Running and Jogging actions.

Table 4.3 reports the action' average accuracy on the KTH dataset. KTH is a relatively simple dataset with almost no background clutter and no camera motion. The videos are indeed recorded in a very constrained and share similar viewpoint and appearance. One specific aspect of this dataset is the high similarity between its *Jogging* and *Running* action as Figure 4-7 displays. Compared to the BoW, table 4.3 shows that cov-max obtains an average accuracy gain of 1.4% for *Jogging* while cov-avg achieves 10.2% gains for the *Running* action. This particular example shows that covariance pooling capture additional discriminative information, compared to a BoW representation that can be exploited during the classification.

4.5.4 Analysis on Real World Video Datasets

Shoot bow Kick ball Shoot gun Swing baseball

Figure 4-6: Illustration of few HMDB actions.

(a) Features under strong motion (b) Features under shot transition

Figure 4-7: Feature clutter illustration on the HMDB dataset.

Figure 4-8 reports the average accuracy per action on the HMDB dataset. Cov-avg obtains the best performance on the action *fencing*, *shoot bow*, *shoot gun*, *kick ball*, *swing baseball*... Figure 4-6 shows that all those actions involve a "external"

object with a specific appearance, they are therefore strongly characterized by both appearance and motion. Average covariance signature describes the linear dependencies between the local descriptors dimensions. Since our local descriptors characterize both appearance through HoG and motion with HoF and MbH, their covariance captures mid-level patterns that characterize both motion and appearance, and which are particularly discriminative on those actions. Yet, cov-avg has lower performance on the *Climb stairs*, *Ride Bike* actions. Those actions are subject to strong camera motion and dynamic backgrounds. As a result, lots of extracted features are actually based on the background and not on the action as Figure 4-7a illustrates. The average covariance matrix robustness therefore suffers from a strong motion and occlusion. Results on the *Talk* and *Chews* actions tend to confirm this observation. Indeed, *Talk* and *Chews* videos are characterized by a lot of shot transitions. Dense trajectories [197] do not handle those shot transition as Figure 4-7b shows (this could be easily fix with a shot segmentation algorithm). It therefore adds a lot of cluttered features and impacts the overall performance. While being discriminative, cov-avg suffers from its limited robustness when videos are subject to strong dynamic background and clutter.

The behavior of the Cov-max signatures is different from cov-avg. Due to the max-pooling; cov-max is more robust than cov-avg. It indeed outperforms cov-avg on the *Climb stairs*, *Ride Bike* and achieves the best-performance on *Talk*, despite its inherent clutter. However, contrary to cov-avg, cov-max does not estimate the sample covariance matrix associated with the descriptor distribution. The statistics associated with the max-operator seems less discriminative as it only slightly outperforms, or even underperforms, the BoW representation on the action *fencing*, *shoot bow*, *shoot gun*, *kick ball* and *swing baseball*. While being more robust, cov-max captures less discriminative information.

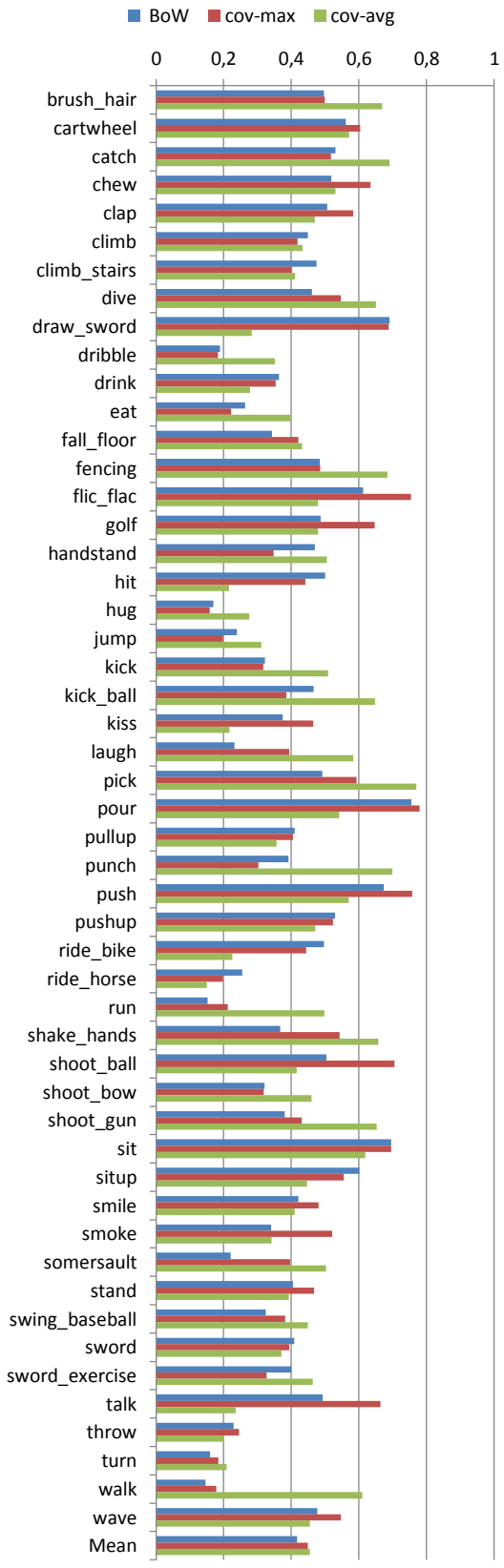


Figure 4-8: Per class average accuracy on the HMDB datasets.

	Track	HoG	HoF	MbH	All
Track	9.5	26.2	31.7	32.1	
HoG	26.2	25.9	37.6	40.2	
HoF	31.7	37.6	30.7	40.7	
MbH	32.1	40.2	40.7	31.5	
All					45.0

Table 4.4: Average accuracies of the intra and inter-descriptor covariances on the HMDB dataset for the max covariance pooling. It clearly highlights that the covariance inter-descriptors is more discriminative than the covariance intra-descriptor. Covariance is therefore especially performant when several descriptors are considered.

We also study the impact of the different descriptors covariance. Table 4.4 reports the average accuracies for the covariance associated with the intra or inter-dimensions. Table 4.3 clearly shows that the feature inter-dependencies information outperforms the features intra-correlation. Indeed, the inter-descriptor covariance performances always outperform the intra-descriptor variance when considered independently. Covariance pooling is therefore particularly efficient due to the use of several local descriptors capturing complementary information.

Finally, we observe the complementarities between the different video representations. To quantify theses complementarities, we consider action lists ranked by the signature performance scores. For each aggregation approach (BoW, cov-avg, cov-max), we denote the action ranking list $\mathbf{p} = \{p_1, ..., p_{51}\}$ where p_i is the rank associated with the ith actions according to the performance score. We compute the Spearman's ρ factor (4.23) for each pooling list pair $\mathbf{p}$ and $\mathbf{p}'$.

$$\rho(\mathbf{p}, \mathbf{p}') = \frac{\sum_{i=1}^{51}(p_i - \bar{p})(p'_i - \bar{p'})}{\sqrt{\sum_{i=1}^{51}(p_i - \bar{p})^2 \sum_{i=1}^{51}(p'_i - \bar{p'})^2}} \tag{4.23}$$

where $\bar{p}$ is the mean ranking value, equal to 26 in case since we perform our evaluation on 51 actions. Table 4.5 shows the spearman ρ factor for the different aggregation schemes. While cov-max is correlated with the BoW representation, cov-avg on the contrary is more independent. It therefore appears that cov-avg captures discrim-

	BoW	cov-avg	cov-max
BoW	1	0.27	0.87
cov-avg	0.27	1	0.29
cov-max	0.87	0.29	1

Table 4.5: Spearman ρ factor of the different aggregation schemes.

inative information which is more complementary to the BoW representation than cov-max.

4.5.5 Is Covariance Matrix Structure Relevant for Classification?

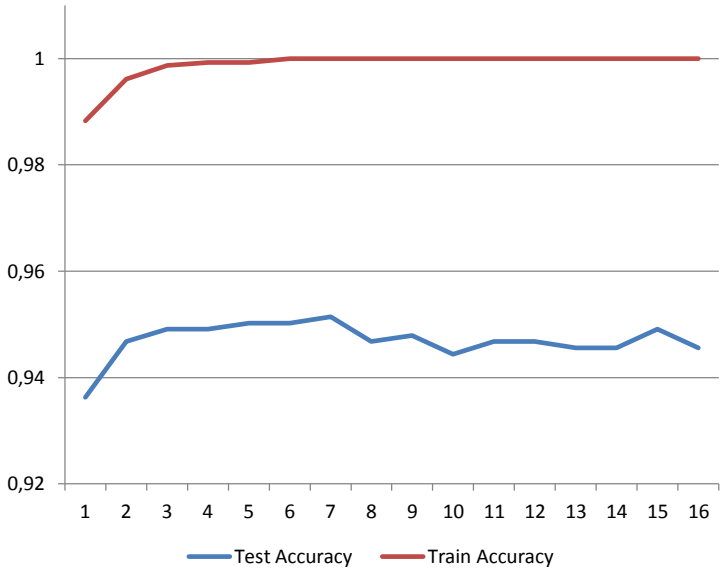
In the second experiment, we aim at determining if the 2D spatial organization of the covariance matrix contains useful information for classification. To leverage the matrix structure, we take advantage of multi-compound bi-linear SVM. For this experiment, we consider the cov-avg signatures.

	Linear	Bi-Linear	Relative Gain
KTH	94.6	**95.1**	0.5%
HMDB	44.5	**48.3**	8.5%

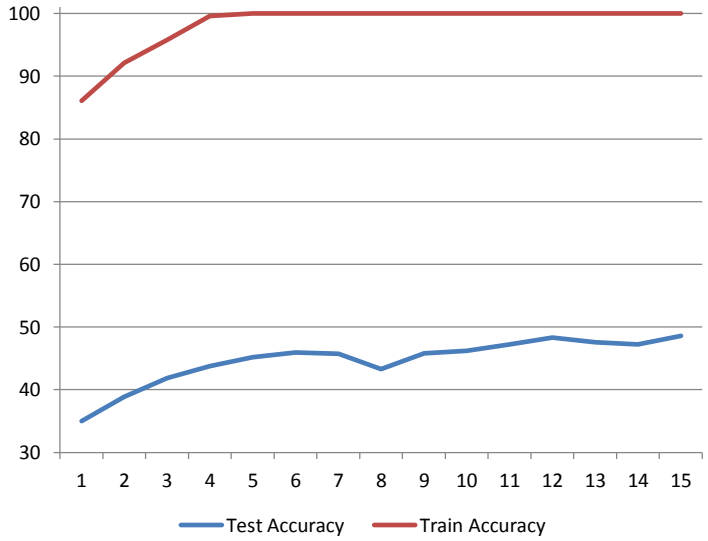
Table 4.6: Average accuracies of linear and bi-linear model for the cov-avg signatures on the KTH and HMDB dataset.

Table 4.6 reports the average accuracy of the linear model and bi-linear model. The bi-linear SVM model uses 7 compound components on the KTH dataset and 15 compound components on HMDB. Table 4.6 shows that the bi-linear model obtains a performance gain on both datasets. Matrix spatial layout therefore contains discriminative information helpful for classification. While a limited gain of 0.5% is obtained on the constrained KTH dataset, an important gain of 9% is achieved on HMDB. Bi-linear is therefore especially useful for large realistic dataset.

Choosing the right number of components for a given dataset is critical to the



(a) KTH



(b) HMDB

Figure 4-9: Test and training accuracies for different compound numbers for the bi-linear SVM applied on the the KTH and HMDB dataset.

classification performance as Figure 4-9 shows. This figure studies the impact of C, the number of compound components, on the classification performance. It reports test average accuracy (the model is learned on training data and tested on testing data) and train average accuracy (the model is learned on training data and also tested on training data).

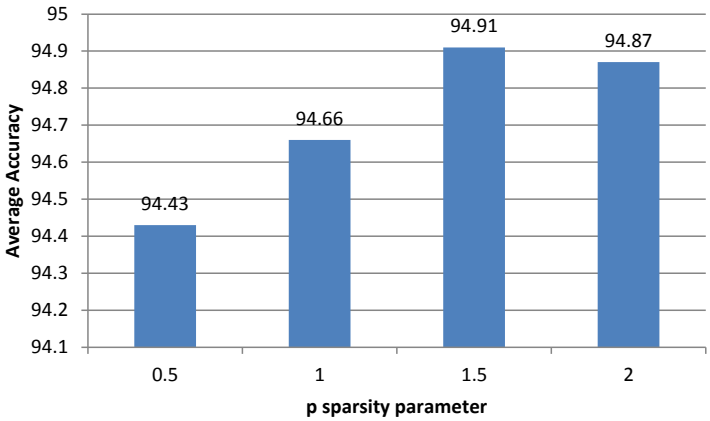
Figure 4-9a focuses on the KTH dataset. When the number of compound components is too small $C < 4$, the bi-linear model doesn't achieve 100% average accuracy on the train data. In this case, the bi-linear model is too simple to capture the data complexity, *i.e.* its VC dimension is too small. By adding compound components, we increase the expressivity of our model and improve the train accuracy up to 100% for $C \geq 4$. The test accuracy also increases with the addition of compound components, from 93.6 for $C = 1$ up to 95.2 for $C = 7$, a relative gain of 2%. However, when $C > 7$, we observe that test average accuracy starts to slowly decrease. It shows that when the number of component becomes too high, the bi-linear SVM model can overfit. The VC dimension of the classification model is to large relatively to the training dataset.

C is also critical for the HMDB performance as shown in Figure 4-9b. By increasing C, the test average accuracy improves by an impressive gain of 38%: from 38.9 for $C = 1$ to 48.3 for $C = 15$. Also, we don't observe any overfiting of the bi-linear model in Figure 4-9b. HMDB training videos are more numerous and diverse than the KTH videos. It therefore requires a model with more compound components to characterize them.

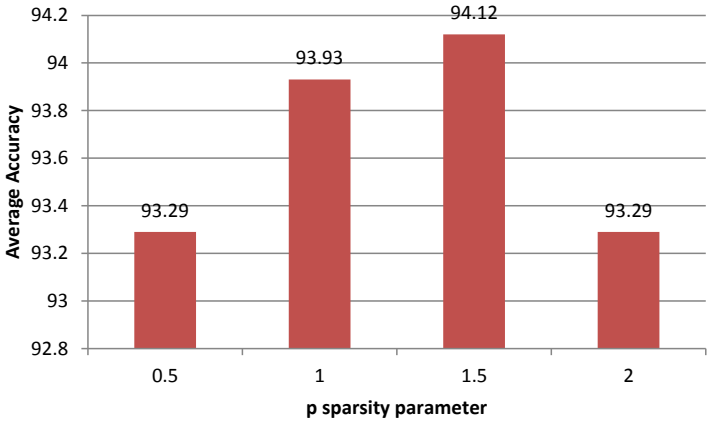
4.5.6 Are Covariance and First-Order Representation Complementary?

In a final experimentation, we want to determine if BoW and covariance representations are complementary. We therefore investigate the combination of BoW and covariance representations using the multiple contexts model as described in section 4.4.3.

Three parameters control our multiple contexts model, the regularizer weight λ, the regularizer sparsity p, the number of bi-linear compound components C. λ is set to 0.1, see section 3.4.2. C is set to 7 for KTH and 15 for HMDB accordingly to section 4.5.5. We study the impact of the sparsity parameter p in the following.



(a) cov-avg+BoW



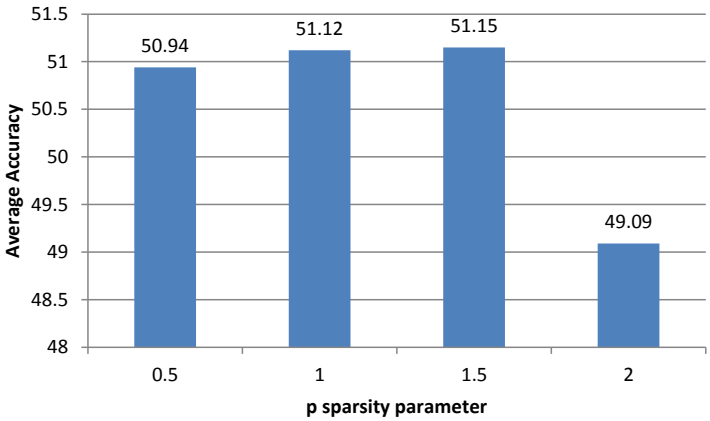
(b) cov-max+BoW

Figure 4-10: KTH dataset.

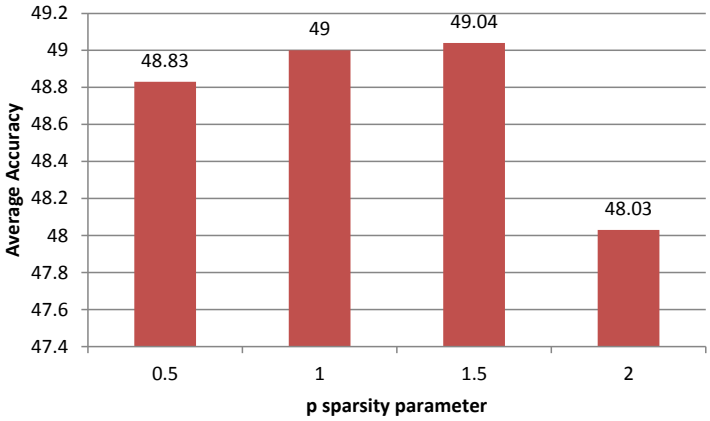
	Best Individual		Best Combination		Relative Gain
KTH	cov-max	95.4	cov-avg+BoW	94.91	-
HMDB	cov-avg (bi-linear)	48.3	cov-avg+BoW	51.1	5%

Table 4.7: Comparaison of individual and combination performances. Average accuracies are reported.

Figure 7-1 reports the average accuracy of the multiple contexts model on the KTH dataset. Figure 4-10a displays the cov-avg and BoW combination for different value of p while Figure 4-10b shows the cov-max+BoW results. On the constrained KTH dataset, the combination of cov-avg+BoW and cov-max+BoW outperforms BoW covariance. However, it does not outperform the individual performance cov-



(a) cov-avg+BoW



(b) cov-max+BoW

Figure 4-11: HMDB dataset.

avg and cov-max performance as Table 4.7 shows. KTH is a constrained dataset, and its actions have limited variability. In this case, covariance representation is discriminative enough to represent the action.

Covariance and BoW combinaition does lead to performance gain on the HMDB dataset. HMDB videos have very diverse appearances, therefore, combining first orders and covariance statistics allows obtaining representations with stronger discriminative ability. cov-avg+BoW combination outperforms cov-max+BoW by a relative difference of 4%. It confirms the Spearman ρ factor results (see Table 4.5) stating that cov-avg is more complementary than cov-max with BoW,

Figure 4-11 shows that inducing sparsity $p < 2$ in our action model does improve the performance for both cov-avg+BoW and cov-max+BoW on the HMDB dataset. Adding some sparsity in our classification model allows emphasizing the representation, BoW or covariance, which fit at best our action. Setting $p = 1.5$ obtains a gain of 4% for cov-avg+BoW, and 2% for cov-max+BoW, compare to a traditional non-sparse SVM, *i.e.* $p = 2$.

4.6 Conclusion

This chapter investigated the inter-dependencies of different local trajectory descriptors (HoG, HoF, MbH, Track) to characterize video. We proposed two main contributions: a low-level video context that captures feature inter-dependency information through their covariation and a bi-linear learning model which classify directly the covariance matrices. Our experimental results showed that:

- covariance between different local descriptors contains discriminative information helpful for classification. Indeed, our covariance representations, cov-avg and cov-max, outperforms the traditional BoW on both constrained or unconstrained dataset up to 8% relatively. Capturing higher-order statistics, covariance allows designing video signatures with strong discriminative ability, but tends to suffer when a lot of outlier features are present;

- a bi-linear model, which takes into account the covariance matrix structure, further improves the classification performance. On HMDB, gain of 8.5% is achieved by the bi-linear model relatively to linear one. We also saw that the number of components composing the bi-linear model is critical to the performance;

- covariance representations are complementary to the BoW. When there are properly combined, a performance gain of 22% is achieved relatively to the sole BoW representation on the unconstrained HMDB dataset;

- due to its robustness, cov-max tends to have better performance than cov-avg, when they are used individually. However, cov-avg is complementary to BoW and their combination leads to better performance than BoW and cov-max combined.

Chapter 5

Task-Specific Space-Time Context

In this chapter, we develop a new space-time context. State-of-art embeds space-time information using predefined and fixed segmentation grids. Consequently, it does not take into account the action space-time layout. By contrast, our approach:

- learns action-adapted segmentation grids directly from the video data;
- infers simultaneously the grids layout and the action appearance model to maximize their joint discriminative capability.

We provide an extensive evaluation of our adaptive grid context on 4 publicly available datasets showing the suitability of our approach. Our solution constantly outperforms the fixed segmentation grids, up to 6%.

5.1 Motivation: Task-Specific Segmentation

In this chapter, we focus our effort on the space-time context modeling. By leveraging the space-time context in our video representation, we aim at benefiting from the space-time discriminative information.

Definition 7. *Space-time contexts: any spatio-temporal information that encapsulates the spatio-temporal layout and transition, relative position, global and semi-local statistics etc, of the low-level visual features*

5.1.1 Local Features Space-Time Context

To achieve invariance toward viewpoint change, traditional bag-of-words representation (BoW) [170] is robust to geometric transformations (translation, rotation and scale). BoW models an image or a video as an orderless collection of invariant local features. It doesn't take into account the feature positions, all the features are pooled globally in the video space-time volume. Features are therefore assumed to be independent from their localizations.

This assumption is clearly false since nearby features, in the space-time domain, are strongly correlated [22]. More importantly, it has been shown that local features positions do convey useful information for extracting semantic descriptions from visual contents [102, 106].

Surf Jetski

Figure 5-1: Surf and Jetski Frame Examples.

Discriminative information is not uniformly distributed in videos as actions do not stretch upon the entire video space-time volumes, but are localized in specific sub-regions. Figure 5-1 displays two video frame examples of a *Surf* and a *JetSki* action along with their localizations. Despite depicting different actions, the two videos have comparable backgrounds composed by sea and buildings. An orderless representation, such as BoW, mixes the foreground and the background statistics and therefore results in two video signatures which have a certain degree of similarity since the video backgrounds are akin in this case.

Orderless representations neglect local feature space-time localization. By taking into consideration the space-time context, we can increase the discriminative power of video signatures. For instance, in the previous example, if we construct one bag-of-words representation using only the local features inside the red-rectangle, we are able to dissociate the action local feature statistics from the background. This simple example highlights the importance of space-time context in video representation.

5.1.2 Space-Time Context in BoW Representation

Previous section shows that taking into account the space-time context could lead to more discriminative representations of videos. Spatial pooling has been introduced [102, 106] to leverage this space-time context. Instead of pooling globally in the space-time volume, this approach pools the local features in local neighborhood which are defined through fixed segmentation grids, see Figure 5-2.

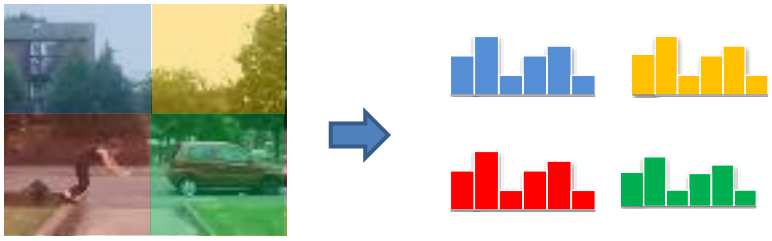


Figure 5-2: Spatial Pooling: the video volume is divided in different space-time cell according to a regular grid, and, one BoW representation is computed inside each cell.

Statement While providing a coarse localization of local features in videos, spatial pooling generally relies on statically defined grids. Since they are predefined, those grids don't necessary fit the local features space-time distribution. In particular, we distinguish two types of action space time contexts: static-space time context where the action localization remains stable through the video and dynamic space-time context which sees the action position evolves with time (see Figure 5-3).

Dynamic space-time context are subject to dramatic space-time variance. In this case, a predefined grid may divide one action across several grid-cells. As for orderless representation, such a grid blends background clutter and action statistics which leads to a significant performance drop. To tackle this issue, we propose to learn the segmentation grid directly from the video data, in order to align the segmentation grid to the local features space-time location.

Static space-time context

Dynamic space-time context

Figure 5-3: Static vs Dynamic space-time context. Static space-time context remains stable over time while dynamic context knows variation.

5.1.3 Our Contribution

This chapter introduces the Adaptive Grids (agBoW) which are task-specific segmentation grids. agBoW learns the space-time shape associated with an action directly from training videos, estimating the action likely positions. Both appearances and structural information are used to determine the usual action localizations in the training videos. Moreover, the segmentation grids are defined in a bottom-up manner. They have an aperiodic geometry, *i.e.* adaptive grids can divide the video space-time volume in a non-regular fashion as highlighted by Figure 5-4.

This latter property is necessary to obtain segmentation grids that fit dynamic action regions in videos since their localizations can drastically shift through time.

Periodic 2x2 Segmentation Grid Learned Adaptive Grid

Figure 5-4: Illustration des fixed versus task-specific grids. Task-specific grid coarsely follows the action through time.

To summarize, this chapter adds the following major contributions:

- we introduce a action-specific spatial pooling formulation;
- we advance a new algorithm for learning segmentation grid;
- we learn simultaneously the action appearance and likely position to maximize their joint discriminative power.

5.2 Related Work

In videos, two main approaches exist to capture the space-time context of local features: self-centered models [49, 92, 179] and spatial pooling [102, 123].

Self-centered approaches [49, 92, 179] capture the local space-time context of an action. Sun [179] relies on a bi-gram to capture features co-occurrence information in local neighborhoods. Gilbert [49] improves over the bi-gram representation by focusing only on the most distinctive co-occurrence given an action. Kovashka and Grauman [92] learn the shape associated to the features neighborhood through hierarchical vocabulary. While learning action-specific context models, all the previous approaches consider the space-time context only at a local level in the videos. Features global localization context also conveys discriminative information. In static images state-of-arts recognition systems exploit the local features layout through spatial pooling. Spatial pooling usually relies on Spatial Pyramid Matching (SPM) [106] where an image is partitioned using increasingly finer cells and BoW histograms are computed independently in each grid cell. Spatio Temporal Grid (STG) [102] is the

	Dimensions	Spatial Information	Layout	Geometry	Application
Sun [179]	2D	local	learn	aperiodic	Image
Kovashka [92]	2D	local	learn	aperiodic	Video
Gilbert [49]	2D	local	learn	aperiodic	Image
Lazebnik [106]	2D	global	fixed	periodic	Image
Laptev [102]	2D + time	global	fixed	periodic	Video
Sharma [166]	2D	global	learn	periodic	Image
Harada [56]	2D	global	learn	periodic	Image
Jia [74]	2D	global	learn	periodic	Image
Cao [26]	time	global	learn	periodic	Video
Our approach	2D + time	global	learn	aperiodic	Video

Table 5.1: Comparison of our approach with state-of-arts.

equivalent of SPM for videos. Several Spatio Temporal Grids are predefined. While providing a coarse localization of the local features, STG don't take into account the video data. Grid layout may therefore not be adaptive enough to fit the local features space-time localization statistics.

In this chapter, we combine the two previous lines of work. We aim at learning action-specific models that capture global information about the feature space-time context. Our goal is to exploit richer spatial and temporal information by learning segmentation grids which are adapted to the actions we want to detect. Recent efforts have also explored the same research direction [26, 56, 74, 166]. However, they have several major differences with our approach. Indeed, most of the previous works focus on the image recognition problem and learn 2D grid [56, 74, 166]. Our approach also takes into account the time dimension to model 3D segmentation grids. To our knowledge, Cao [26] has proposed the only task-adapted pooling in video. However, his approach focuses on modeling only the temporal context and discards the spatial information. In addition, the previous approaches tend to focus on segmentation grid with periodic geometry. Jia [74] relies on sparsity to select periodic segmentation grids in an overcomplete basis while Harada [56] learns the weights associated with predefined segmentation grids. Krapack [93] models the local features dispersion of using a Gaussian Mixture Model (GMM) coupled with a Fisher Kernel [93] for static

images. Such an approach captures the feature localization independently from the appearance and motion in one dedicated video signature.

Table 5.1 synthesizes the difference of our approach with the related works. Our approach general principle is detailed in the next section.

5.3 Leveraging Viewpoint Repeatability to Learn Task-Specific Segmentation Grid

Tacit rules guide the making of videos. Movies, for instance, respond to an implicit visual language. In movie videos, camera viewpoint and motion, among other cinematic elements, are controlled by the film director. Each cinematic element has a particular impact on the movie perception, they define a visual language which is used to express a narrative story [124]. Since directors tend to employ same language elements to represent similar situation, we can observe a visual consistency between the different actions. Figure 5-5 shows that a *Talk* action likely involves a close-up face view with movie characters shot from a particular orientation to give a sense of the dialogue exchange. *Run* can be recorded such as the running character face to emphasize the character effort, or from a side view the camera, so the viewer can easily appreciate the running speed and distance. In a same way, *Kiss* action is often located at the center of the screen. A viewpoint repeatability therefore exists between the videos representing a same action.

Viewpoint repeatability is also present in unconstrained and non-edited videos such as online amateur clips. For instance, most of the top-videos on the YouTube website, corresponding to the "Soccer Juggling" request, contain a soccer player localized approximately near the center of each frame. Such a viewpoint is naturally chosen since it highlights the juggling prowess through the video.

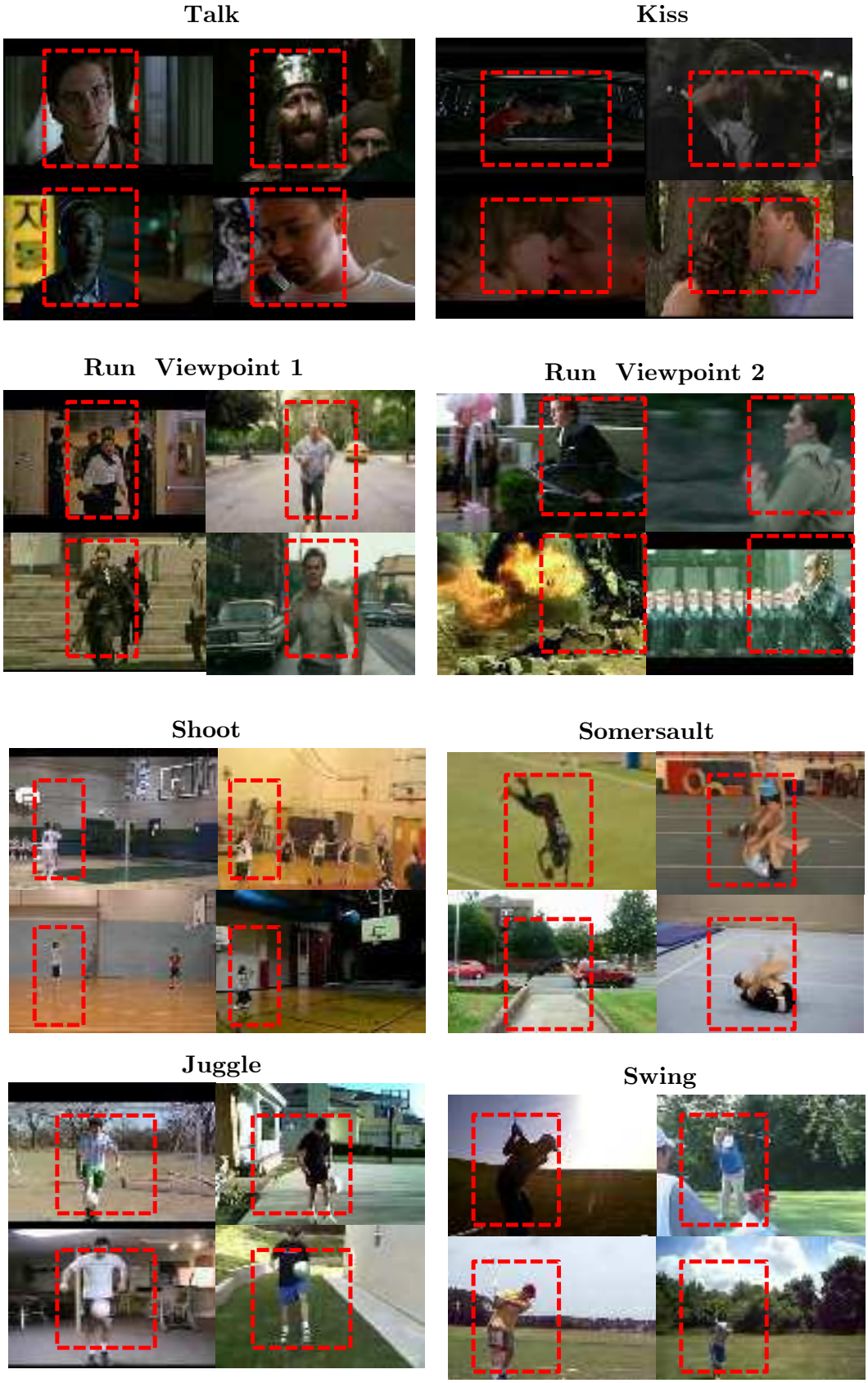


Figure 5-5: Illustration of viewpoints repeatability. Red rectangles indicate the video discriminative regions.

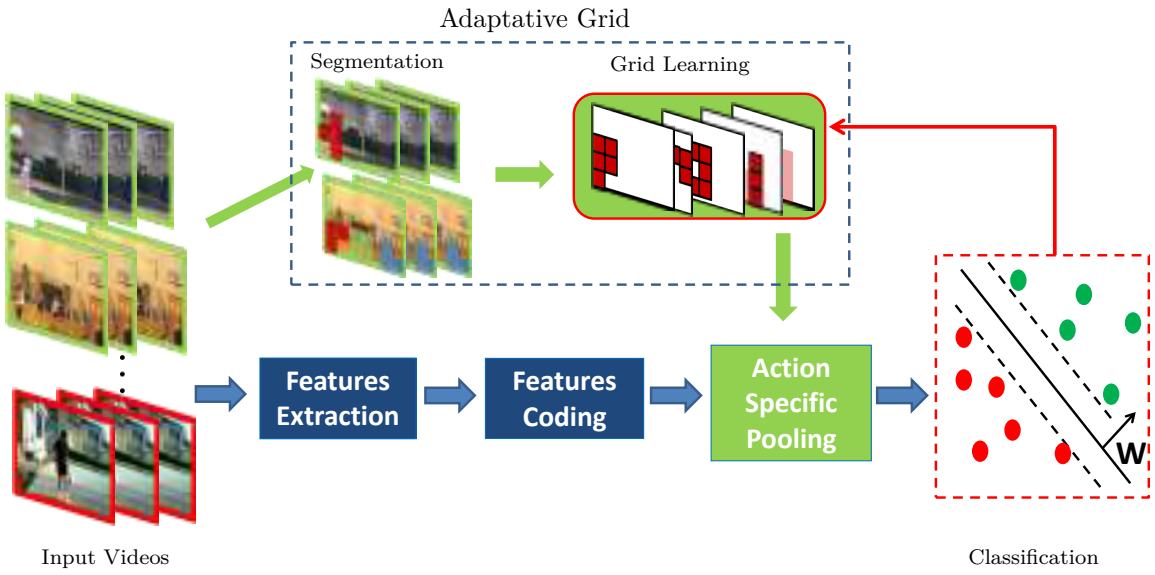


Figure 5-6: Synopsis of action-specific-based recognition: green blocks correspond to our contributions.

Due to the viewpoint repeatabilty, we can observe that the localization of an action is repeatable across different videos. Space-time regions corresponding to the foreground action are likely to have repeatable positions while external elements are inclined to variation. For instance, the action "Soccer Juggling" can be coarsely decomposed in two regions characterizing either the "human leg" or the "soccer ball". Since, most of "Soccer Juggling" YouTube videos contain a soccer player localized approximately near the center of each frame, the "human leg" and the "soccer ball" regions are likely to share similar positions in different videos. On the other hand, background may not be repeatable in different videos. "Soccer Juggling" videos are rarely recorded at the same time and place. The positions of external elements, such as other soccer players, are prone to variation from one video to another.

By learning the region repeatable positions in an action training videos, we can estimate the action usual localization and design action-specific segmentation grids. Figure 5-6 shows our approach. Our grid learning algorithm leverages both video structural and appearance information.

Definition 8. *Motion regions: spatially connected region sharing a homogeneous*

motion.

Structural information is used to compute motion regions having repeatable positions in videos. A motion region is a video space-time sub-volume which shares a homogeneous movement. A segmentation algorithm is used to extract motion regions from the training videos. An unsupervised clustering algorithm is then applied to retain only the regions which appear several times across the training videos.

Appearance information is also used to weight space-time regions that are discriminative relatively to the action model. We take advantage of the action classifier feedback to maximize the importance of discriminative grids, *i.e.* regions which captures relevant information for the action-classifier are emphasized.

Our task-specific context modeling is detailed in the remaining of the chapter. We first introduce a spatial pooling formulation that looses the grid definition from the pooling operation to allow the use of action specific grids. The extraction of structural information from videos is presented in a second time. We then describe our task-specific context model. An extended evaluation of our approach is finally provided.

5.4 Action Specific Pooling

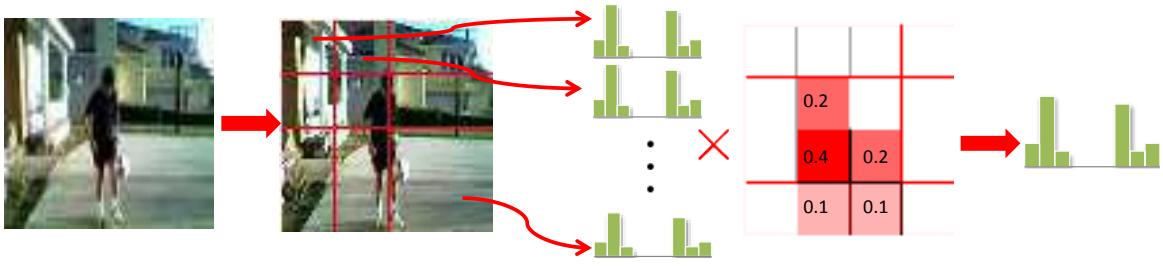


Figure 5-7: Illustration of action-specific pooling.

In this section, we introduce a new formulation to the spatial pooling operation [102, 106] which separates the segmentation grid definition from the actual pool-

ing. The pooling formulation being more flexible, different segmentation grids can then be used to build a video signature. Our formulation can use different segmentation grids depending on the action to recognize. As Figure 5-7 shows, three steps composed the action-specific pooling:

- a video is first divided in several cuboids;
- the local feature distributions inside each cuboid are then captured to obtain a generic intermediate representation;
- the generic intermediate representation is combined with an action-specific grid to obtain our final representation.

A grid is therefore expressed in a bottom-up manner through a linear combination of cuboids.

5.4.1 Generic Intermediate Representation

Let's consider a set of local feature descriptors extracted from a video $\mathbf{D} = \{\mathbf{d}_i\}_{i\in[1,M]}$, and $\mathbf{C} = \{\mathbf{c}_j\}_{j\in[1,L]}$, L space-time cuboids dividing a video. To benefit from video space-time information, local features are pooled locally in each cuboid. The feature distribution of a cuboid $\mathbf{c}_j$ is obtained by computing the first order statistics of the coded features present in the cuboid:

$$\mathbf{X}_j = \max_{\mathbf{d}_i\in\mathbf{c}_j} code(\mathbf{d}_i). \tag{5.1}$$

Here, $\mathbf{d_i} \in \mathbf{c}_j$ is the set of local features which are spatially contained in the cuboid $\mathbf{c}_j$. The function $code : \mathbf{D} \rightarrow \mathbb{R}^K$ is a local feature coding scheme such as hard-coding, sparse-coding or locality coding. L different cuboids lead to L distribtution vectors $\{\mathbf{X}_j\}_{j\in[1,L]}$, with $\mathbf{X}_j \in \mathbf{R}^K$. All the distribution vectors are row-concatenated resulting in a matrix $\mathbf{X} \in \mathbf{R}^{L\times K}$ which describes our video:

$$\mathbf{X} = \begin{bmatrix} x_{1,1} & \dots & x_{1,K} \\ & \dots & \\ x_{M,1} & \dots & x_{L,K} \end{bmatrix}. \tag{5.2}$$

The $\mathbf{X}$ j-th line describes $\mathbf{c}_j$ cuboid:

$$\mathbf{X}_j = [x_{j,k}]_{k\in[1,K]}. \tag{5.3}$$

5.4.2 Action-Specific Intermediate Representation

$\mathbf{X} \in \mathbb{R}^{LxK}$ is a generic intermediate representation which depicts the general space-time context in a video. It captures the features distribution in different space-time cuboids, each space-time cuboid highlights a specific region in the video volume. Due to the viewpoint repeatability, we saw that space-time regions have varying importance depending on the action to recognize (section 5.3). A generic video representation $\mathbf{X} \in \mathbb{R}^{LxK}$ is therefore combined with action-specific segmentation grids to provide an action-specific intermediate representation. Such grids emphasize the space-time regions which are usually discriminative for an action.

We denote by $\mathbf{g} = [g_j]_{j\in[1,L]} \in \mathbb{R}^L$ a grid descriptors where g_j is grid strength response associated with the c_j cuboid. Given this definition, spatial max-pooling, denoted by the operator $\otimes$, can be expressed as:

$$\mathbf{g} \otimes \mathbf{X} = [f_i]_{i\in[1,K]} \text{ with } f_i = \max_{j\in[1,L]} g_j \times \mathbf{X_{j,i}}. \tag{5.4}$$

The video representation $\mathbf{g}\otimes\mathbf{X}$ are then ℓ_2 normalized. (5.4) is a generalization of the traditional spatial pooling [102, 106]. Indeed, if $\mathbf{g}$ defined a set of fixed segmentation grids dividing the video volume with increasingly finer cells, (5.4) corresponds to traditional spatial pooling. Differently, our goal is to infer $\mathbf{g}$ directly from video data, we aims at learning one specific $\mathbf{g}$ capturing the space-time shape of each action.

5.5 Identifying Informative Regions in Videos

Action-specific segmentation grids are constructed by identifying space-time motion-regions which have repeatable positions among different videos. This section describes how to extract and localize such regions in videos, as Figure 5-8 shows.

Actions are highly dynamics in nature. Video regions with non-negligible motion are therefore likely to contain useful cues to characterize those actions. We introduce a motion based segmentation algorithm to identify motion regions in videos. We then propose a descriptor that characterizes the motion region positions.

5.5.1 Region Extraction

Figure 5-8: Synopsis of motion region positions extraction.

To identify space-time regions with non-negligible motion, a segmentation algorithm based on long term point trajectory clustering, inpired from Brox and Malik [23], is applied. The following provides a general description of the algorithm. The complete segmentation algorithm details are given in Appendix A.

We consider a set of dense trajectory features $\mathbf{T} = \{\mathbf{t}_i\}_{i\in[1,M]}$ extracted from a video [196]. Our goal is to compute a set of trajectory clusters $\mathbf{O} = \{\mathbf{o}_j\}_{j\in[1,Q]}$, where each $\mathbf{o}_j$ is a subset of $\mathbf{T}$ that respects some motion and spatial locality constraints.

Q, the number of space-time regions having consistent motion in a video, is not known and can change from one video to another. To cope with this variability, we model our trajectory clustering using a Gibbs point process [10]. A Gibbs model can be seen as a Markov Random Field (MRF) extension, dealing with a varying number of observations [36].

A Gibbs Point process is defined by an unormalized density $h(\mathbf{O}) = e^{-U(\mathbf{O})}$. U is the energy associated with the cluster realization $\mathbf{O}$. It is modeled as a linear

combination of an attraction and repulsion potentials:

$$U(\mathbf{O})) = \sum_{\mathbf{j}\in[\mathbf{1},\mathbf{Q}]} \alpha_1 a(\mathbf{o}_j) + \alpha_2 r(\mathbf{o}_j) \tag{5.5}$$

where

$$a(\mathbf{o}_j) = \sum_{\mathbf{t}_i\in\mathbf{o}_j} \sum_{\mathbf{t}_k\in\mathbf{T}\setminus\{\mathbf{o}_j\}} e^{-\lambda d(\mathbf{t}_i,\mathbf{t}_k)}, \qquad r(\mathbf{o}_j) = \sum_{\mathbf{t}_i\in\mathbf{o}_j} \sum_{\mathbf{t}_k\in\mathbf{o}_j} 1 - e^{-\lambda d(\mathbf{t}_i,\mathbf{t}_k)}. \tag{5.6}$$

In (5.5), α_1, α_2 are potential fusion coefficients that are empirically determined. Here, d is a function measuring the space-time divergence between two trajectories:

$$d(\mathbf{t}_i, \mathbf{t}_j) = \frac{1}{|l-f|} \sum_{k=f}^{l} |\mathbf{p}_k^{\mathbf{t}_i} - \mathbf{p}_k^{\mathbf{t}_j}||\mathbf{m}_k^{\mathbf{t}_i} - \mathbf{m}_k^{\mathbf{t}_j}|. \tag{5.7}$$

In (5.7), we denote by f (respectively l) the first (respectively last) common frame between $\mathbf{t}_i$ and $\mathbf{t}_j$. $\mathbf{p}_k^{\mathbf{t}_i}$ is the $\mathbf{t}_i$ trajectory position at the frame k. $\mathbf{m}_k^{\mathbf{t}_i}$ is the trajectory motion vector. λ is a constant set to 0.1. s ensures that only trajectories spatially close and with similar motion have low divergence.

We minimize (5.5) with the Metropolis-Hasting-Green (MHG [47]) algorithm to obtain trajectory clusters. MHG performs iterative updates $\mathbf{C}$. We successively merge or split clusters based using a normalized minimum cut criteria [10].

5.5.2 Position Extraction

We consider a cluster $\mathbf{o}_i$ computed using the motion segmentation algorithm. As Figure 5-10 shows, the cluster positions are quantized using a segmentation grid to obtain its average localization in the video space-time volume.

A regular grid composed by L cuboids is applied to tile the video. A histogram, counting the number of trajectories falling in each grid cuboid, is then computed. It results in our descriptor $\mathbf{s}_i = [s_j]_{j=[1,L]}$. Since the trajectories are defined on long temporal windows, they are associated with several cuboids. $\mathbf{s}_i$ histogram is finally

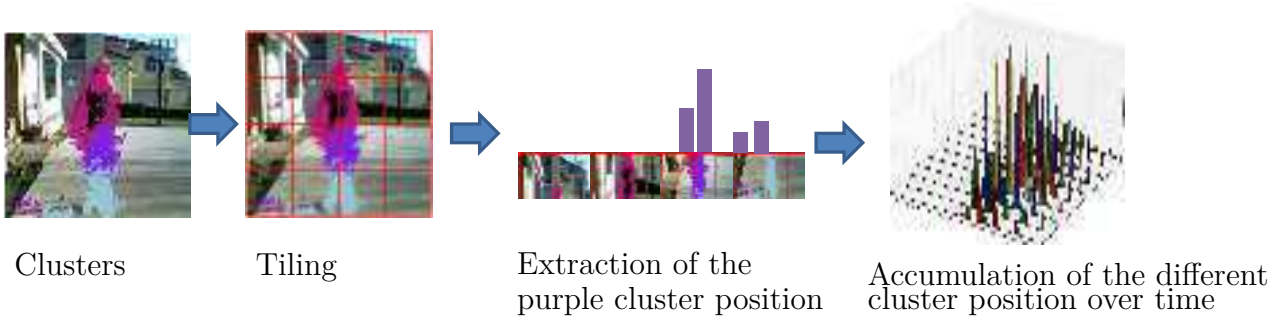


Figure 5-9: Cluster Position Extraction.

ℓ_2 normalized.

5.6 Task-Specific Space-Time Context Modeling

Let $\mathbf{X} = \{\mathbf{X}_i\}_{i\in[1,N]}$ be N generic intermediate representation and $\mathbf{Y} \in \{0,1\}^N$ their corresponding binary labels indicating the presence or absence of an action. Each $\mathbf{X}_i \in \mathbb{R}^{L\times K}$ is an intermediate representation (5.2) which captures the local feature distributions in L different grid cuboids. Our goal is to learn an action-model $(\mathbf{W}, b)$ which captures the correlation between the intermediate representations $\mathbf{X} = \{\mathbf{X}_i\}_{i\in[1,N]}$ and their corresponding labels from the training data $\mathbf{Y} \in \{0,1\}^N$.

We saw in section 5.3 that actions tend to be constrained by a set of repeatable viewpoints. We want to take advantage of this repeatability. We therefore propose to learn Adaptive Grids set, $\mathbf{G} = \{\mathbf{g}_c\}_{c=[1,C]}$, which depicts the space-time layout of each action. Rather than learning an action-model $(\mathbf{W}, b)$ between a label $\mathbf{Y}_i$ and a generic intermediate representation $\mathbf{X}_i$, we benefit from the action specific space-time context and infers the correlation between the label $\mathbf{Y}_i$ and the action-specific representation $\mathbf{g}_c \otimes \mathbf{X}_i$. Each Adaptive Grid is a segmentation grid which captures the space-time shape associated with an action.

In the following we describe how to learn the adaptive segmentation grid $\mathbf{G}$. Our learning algorithm benefits from both structural and appearance information.

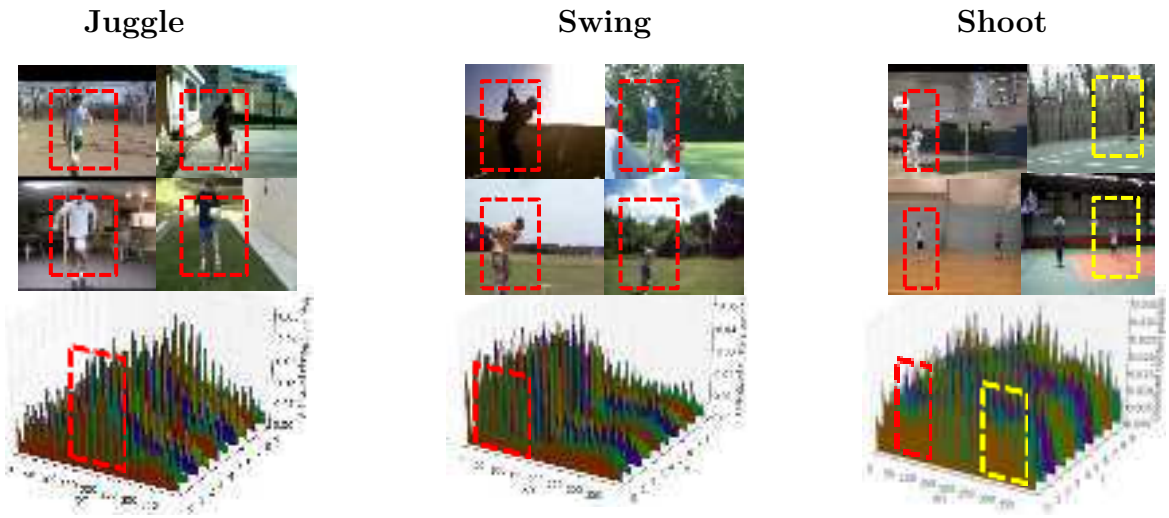


Figure 5-10: Average segmentation grids of the segmentation regions of the "Shoot, "Swing" and "Juggle" action on the YouTube dataset. A 20x20x10 **regular grid is used to quantize the segmented region positions.**

5.6.1 Leveraging Structural Information

We consider a set of video-specific segmentation grids, $\mathbf{S} = \{\mathbf{s}_i\}_{i\in[1,Q]}$, computed using the motion segmentation algorithm.

Figure 5-10 shows the average video-specific segmentation grid for the "Golf", "Shoot" and "Juggle" actions on the YouTube dataset. It shows that most of the motion regions for the "Golf" action, take place on the left corner of the video. Motion regions of "Juggle" happen most of the time in the middle of the video. Motion regions for the *Shoot* action are more equally distributed in the space-time since this action is represented using multiple viewpoints.

Figure 5-10 highlights that the repeatable localizations of motion regions tend to correlate with the action localization. Therefore, by identifying the position of repeatable segmented regions in videos, we can obtain a coarse usual localization of the action of interest.

To identify the repeatable patterns in $\mathbf{S}$, we construct $\mathbf{G}$ using "bag-of-grids" model. Task-specific are learned as a codebook of $\mathbf{S}$ by minimizing the mean square

reconstruction error:

$$\hat{G} = \underset{\mathbf{G}\in\mathbb{R}^{\mathbf{C}\times\mathbf{L}},\alpha\in\mathbb{R}^{C\times Q}}{\arg\min} ||\mathbf{G} - \alpha\mathbf{S}||_2^2. \tag{5.8}$$

In (5.8), $\alpha \in \mathbb{R}^{C\times Q}$ associates the video-specific segmentation grid of S to corresponding Adaptive Grids. We use k-means algorithm to optimize (5.8). An Adaptive Grid is a grid word $\mathbf{g}_i \in \mathbb{R}^N$ which is fuzzy and has aperiodic geometry by construction. Several Adaptive Grids are learned for one action since an action can be represented through multiple viewpoints.

5.6.2 Leveraging Appearance Information

The assumption that motion regions having repeatable positions are part of the action foreground can be too strong for some action. Due to some clutter or camera motion, some background regions can also have appears at similar positions in different videos. While those regions don't capture information relative to the action of interest, they impact the Adaptive Grids computation. To tackle this issue, we leverage the video appearance information. We learn the task-specific grids $\mathbf{G}$ and the action model $(\mathbf{W}, b)$ simultaneously.

We consider a linear model $\mathbf{W} = \{\mathbf{W}_c\}_{c=[1,C]}$ with its bias term $b \in \mathbb{R}$. $\mathbf{W_c} \in \mathbb{R}^d$ is the group of $\mathbf{W}$ linear coefficients correlating with the Adaptive Grid $\mathbf{g}_c$. Our primal learning objective function has the following form:

$$E(\mathbf{G}, \mathbf{W}, b) = \sum_{i=1}^{N} L(\mathbf{Y}_i, \mathbf{X}^i) + \lambda\Omega(W) + \gamma\Gamma(G), \tag{5.9}$$

$$L(\mathbf{Y}_i, \mathbf{X}^i) = \max(0, \mathbf{Y}_i(\sum_{c=1}^{C}(g_c \otimes \mathbf{X}_i)\mathbf{W}_c + b))^2. \tag{5.10}$$

Here, $g_c \otimes \mathbf{X}_i$ is the action-specific pooling operation (5.3). L is the square hinge

Algorithm 4 Concave-Convex Learning.

Input: Input data $\mathbf{X} \in \mathbb{R}^{N \times d}$, labels $\mathbf{Y} \in \{0,1\}^N$, segmentation grids $\mathbf{S} \in \mathbf{R}^{N \times M}$ Parameters λ, Ω, α

Output: $\mathbf{W} \in \mathbb{R}^d, b \in \mathbb{R}$ and $\mathbf{G} \in \mathbf{R}^M$

1: Initialize $\mathbf{W} \in \mathbb{R}^d$ and b at random;
2: Solve $\arg\min_{\mathbf{G},\alpha} ||\mathbf{G} - \alpha\mathbf{S}||_2^2$ with k-means;
3: **repeat**
4: L-LBFGS on $(\mathbf{W}, b)$;
5: Stochastic gradient on $\mathbf{G}$;
6: **until** Convergence

loss (6.15). Ω is the action model regularizer:

$$\Omega(\mathbf{W}) = \|\mathbf{W}\|_2^2, \tag{5.11}$$

while Γ is the Adaptive Grids regularizer which is expressed as the square reconstruction error:

$$\Gamma(\mathbf{G}) = ||\mathbf{G} - \alpha\mathbf{S}||_2^2. \tag{5.12}$$

In (5.9) Γ constraints $\mathbf{G}$ based on the k-means clustering. Note that α is determined beforehand through minimization of (5.8). The objective function (5.9) therefore depends on both structural and appearance information. Indeed, we embed structural information in our objective function through the Γ regularizer. Our goal is to infer segmentation grids $\mathbf{G}$ that fit $\mathbf{S}$ while having a discriminative appearance. γ is a trade-off parameter weighting the structural information importance and the appearance discriminative power.

5.6.3 Optimization

To minimize (5.9) according to $\mathbf{G}$ and $(\mathbf{W}, b)$, we apply concave-convex procedure (5). By fixing alternatively $\mathbf{G}$ and $(\mathbf{W}, b)$, we iteratively update our model. With $\mathbf{G}$ fixed, (5.9) becomes a classic linear SVM problem which is minimized directly by using

Algorithm 5 Stochastic Gradient Descent on G.

Input: Input data $\mathbf{X} \in \mathbb{R}^{N \times d}$, labels $\mathbf{Y} \in \{0,1\}^N$, segmentation grids $\mathbf{S} \in \mathbf{R}^{N \times M}$, $\mathbf{G} \in \mathbb{R}^d$ Regularization parameters λ, ω

Output: $\mathbf{G} \in \mathbb{R}^d$

Set learning rate γ
repeat
  Select i at random in $[1, ..., N]$
  if $\mathbf{Y}_i(\sum_{j=1}^{C}(g_j \otimes \mathbf{X}^i)\mathbf{W}_j + b) \geq 1$ **then**
    $\mathbf{H} = \max_k \mathbf{X}_k^i$
    $\frac{\partial E}{\partial \mathbf{C}} = 2(\mathbf{YHW} - (\mathbf{GH}^{\cdot 2} + b \times \mathbf{H})) + \omega(2\mathbf{G}\alpha\alpha^T - \alpha\mathbf{S})$
  else
    $\frac{\partial E}{\partial \mathbf{C}} = \omega(2\mathbf{G}\alpha\alpha^T - \alpha\mathbf{S})$
  end if
  $\mathbf{G} = \mathbf{G} - \gamma\frac{\partial E}{\partial \mathbf{C}}$
until Convergence

a classic quasi-Newton Limited-memory-BFGS optimization algorithm. When $(\mathbf{W}, b)$ is fixed, (5.9) is not smooth anymore due to the max-pooling operation. Instead of a Limited-memory-BFGS, we rely on a stochastic gradient descent (5) which has demonstrated good results with this type of optimization problem [165].

5.7 Relevance of Adaptive Grids: Evaluation

This section provides an evaluation of the Adaptive Grids (agBoW) context. We first introduce our experimental setting.

5.7.1 Experimental Setting

To evaluate the impact of the Adaptive Grids representation (agBoW), we compare our approach with an orderless bag-of-words representation (BoW), and, a fix grid pooling based bag-of-words (spBoW). BoW discards the space-time context information while spBoW embeds space-time information relying on fixed and predefined segmentation grids. Most of the settings are identical to the one used for the covariance experimentation (section 4.5.1).

To build our bag-of-words representation (BoW), we rely on dense-trajectory fea-

tures encoded with LLC coding [114] and max-pooling. Spatial pooling based bag-of-words (spBoW) is computed by pooling the features in local space-time neighborhood rather than globally in the video volume. A video is divided using a regular segmentation grid and the local features are pooled in each grid cell independently [102]. The resulting histograms are then ℓ_2-normalized and concatenated. When it is not specified otherwise, spBoW uses a $2 \times 2 \times 2$ and $3 \times 3 \times 3$ segmentation grids. Such a grid divides each video dimension in respectively two or three cells. It has been shown that those grids achieve good performances on several datasets [197]. To learn agBoW, we initially divide our video into 4x4x4 exclusive cuboids. 16 AG are learned for each action. An evaluation of the AG parameters is provided in the following.

The relevance and suitability of our approach is shown on 4 publicly available datasets: UT-Interaction 1 and 2, UCF-YouTube, and HMDB [95, 113, 160] which are described in section 2.3.

5.7.2 Does the Space-Time Context Relevant Help in Improving Action Annotation?

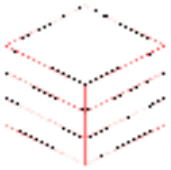
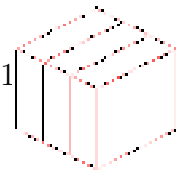
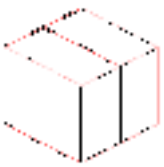
In a first experiment, we evaluate the impact of the space-time context on the action recognition performance, and, show the need for task-specific segmentation for action annotation. We evaluate several fixed grids with predefined regular geometry on the challenging HMDB dataset. We consider several segmentation grids which divide the x, y and t dimensions in 2 or 3 cells, as shown in Figure 5-11, and compare them to an orderless BoW representation.

Table 5-11a reports the average accuracy score for the different fixed segmentation grids and the orderless representation. We first observe that 5 out of the 6 fix segmentation grids outperform the BoW representation, up to 10.6%. Space-time context does convey discriminative information for action recognition. However, the performance gain depends on the grid segmentation layout. A 1x1x3 grid decreases the performance by 4% comparatively to the BoW, while a 1x3x1 grid obtains a 10.6%

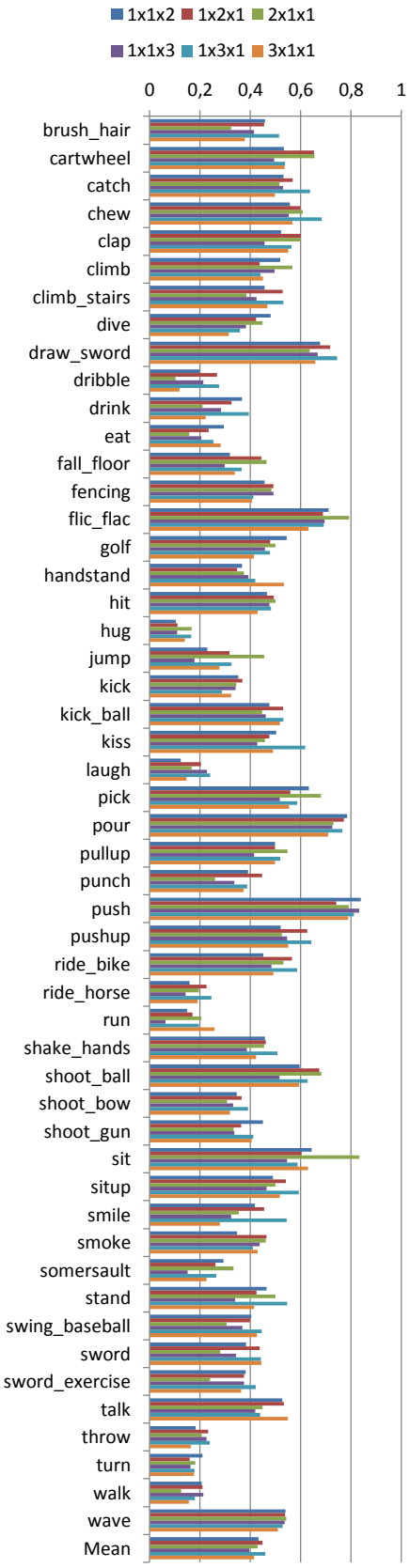
performance gain. A 1x3x1 segmentation grid therefore provides a better modeling of the space-time context than a 1x1x3 grid on average.

Figure 5-11b reports the accuracy score obtained by the different segmentation grid on each action. It shows that the optimal segmentation layout is action dependent. For instance, a 2x1x1 segmentation grid obtains the best performance for the actions *fall floor*, *flic flac* and *hug* while the 3x1x1 grid reaches the best results for *pick*, *pour* and *ride horse*. This observation is also verified in Table 5-11a which, reports the number of actions reaching the best classification performances for each segmentation grid. No grid constantly outperforms the other segmentation schemes. Despite having the lowest performance on average, a 1x1x3 still obtains the best scores for 3 actions.

We conclude from this first experiment shows that (1) space-time context provides discriminative information; (2) space-time context is action-dependent, *i.e.* some grids are better fit to capture the action space-time contexts. By learning action-adaptive segmentation grids directly from the data, we hope to leverage the optimal segmentation layout for each action.

Grid Layout	Acc	#A
BoW -	41.6	4
2x1x1 	42.9	10
3x1x1 	41.4	3
1x2x1 	44.8	4
1x3x1 	46.0	20
1x1x2 	43.1	7
1x1x3 	39.6	3

(a) Fixed segmentation grids evaluation. Acc is the average accuracies while #A is the number of action reaching the best classification scores using this particular segmentation grids.



(b) Accuracy Per Action

Figure 5-11: Evaluation of fixed regular grids on HMDB.

5.7.3 Adaptive Grids: Proof of Concept

In a second experiment, we evaluate our agBoW and compare it to traditional BoW and spBoW (with a 2x2x2 segmentation grid). We consider video datasets with constrained complexity to ease the agBoW analysis. We choose to test our approach on the UT-interaction 1 (UT-1) and UT-interaction 2 (UT-2) datasets [160] (see Section 2.3.1). They are both composed by 5 actor-actor interactions (*HandShake*, *Hug*, *Kick*, *Punch* and *Push*) and one actor action (*Point*). Each action-class has 10 videos. UT-1 and UT-2 videos have mostly static backgrounds are subject only to small camera jitter. While interaction classes know a strong localization variation through the video *Point* action localization remains stable over time. Since, interaction classes exhibit a dynamic and complex space-time context, we choose UT-1 and UT-2 datasets to show that agBoW is able to capture the complex space-time shapes of an action.

UT-Interaction 1 and 2 are rather small datasets, hence we learn one adaptive segmentation grid per action to avoid overfitting. Indeed, only 9 positive video examples are available to learn a model for an action.

Table 5.2 reports the average accuracies of the different representations. It shows that agBoW outperforms the BoW and spBoW by respectively 6% and 12% on UT-interaction 1; 10% and 8% for UT-interactions 2. Modeling of an action-specific space-time context does help for action recognition.

Quite surprisingly, the spBoW obtains worst results than BoW on UT-Interaction 1, spBoW shows a performance drop of 5%, most of the UT-1 and UT-2 actions have dynamic space-time context, the actions are subject to large localization variance through time. For instance, the two human involve in the action *Shake-Hand* are first located on the video left and right border, and then, move toward the video center to perform the hand shaking as Figure 5-12 shows. Due to its static definition, a spBoW does not handle the action localization variability, an action will be divided by across different grid cells leading to a representation that confuses foreground and background information.

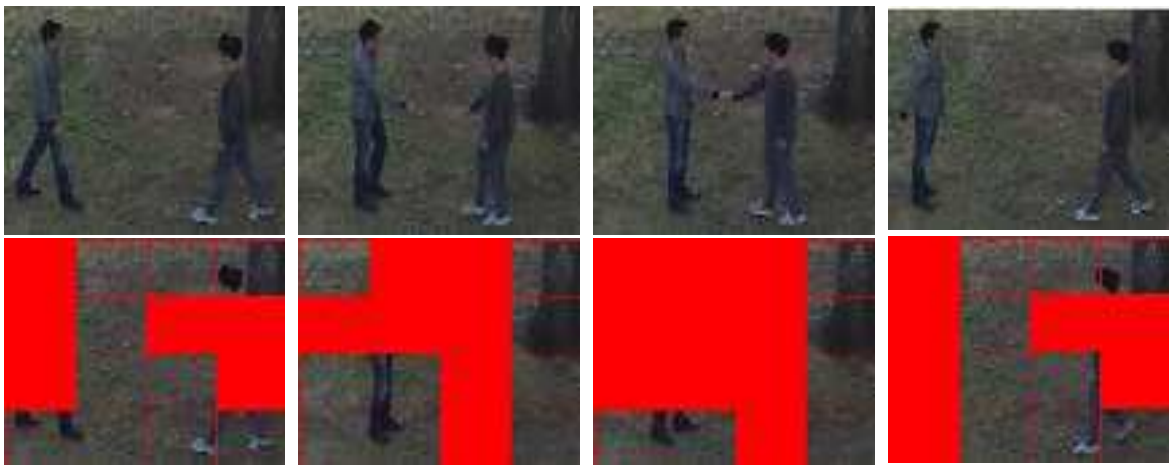
	HandShake	Hug	Kick	Point	Punch	Push	**Mean**
BoW	90	100	90	100	55	80	85.8
spBoW	95	100	90	100	40	70	81.6
agBoW	**100**	**100**	**100**	**100**	**70**	**80**	**91.7**

(a) Set1

	HandShake	Hug	Kick	Point	Punch	Push	**Mean**
BoW	100	**100**	90	95	50	80	85.8
spBoW	90	95	95	100	65	80	87.5
agBoW	**100**	95	**100**	**100**	**80**	**95**	**95**

(b) Set2

Table 5.2: Average Accuracy on UT-Interaction 1 and 2.



T=1 T=2 T=3 T=4

Figure 5-12: Adaptive Grid learned for *Shake-Hand*. First line: frame examples of a *Shake-Hand* action sampled at different time in a video. Second line: 4x4x4 Adative Grid learned for the *Shake-Hand* action. Only grid the cuboids with a reponse strenght superior to 0.1 **are displayed.**

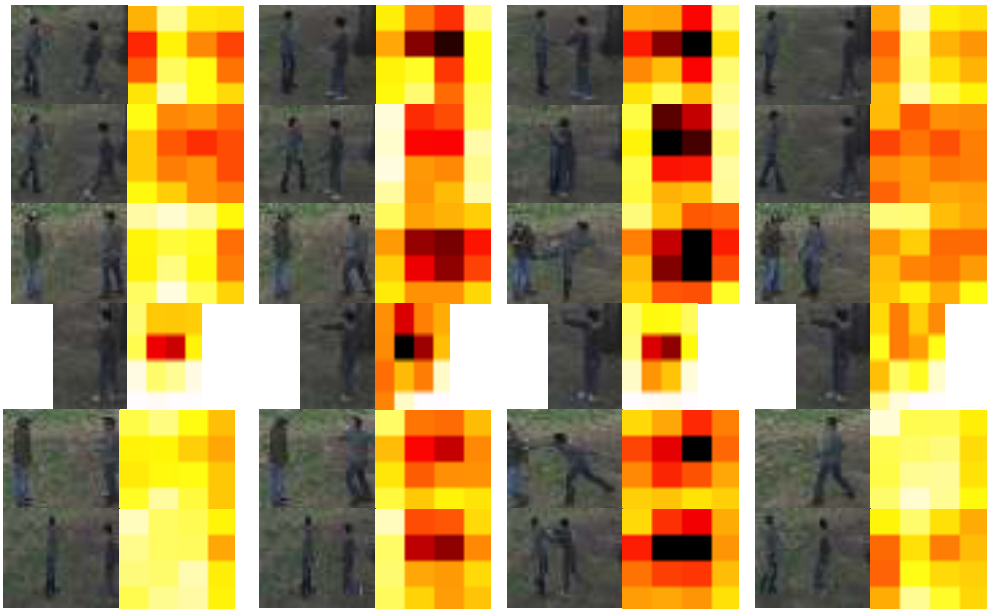


Figure 5-13: 4x4x4 Adaptive Grids on UT-interaction 2 dataset. Each line correspond to one UT-interaction 2 action. In each column, the first image shows a video example sampled at different time, the second image displays the heat map of the action Adaptive Grid.

Differently, agBoW learns directly the segmentation grids from the video data. It takes into consideration the video content to build segmentation grids that fit the local features space-time distribution. As a result, it learns segmentation that coarsely follows the action through time (see Figure 5-12). On *Shake-Hand*, agBoW reaches an average accuracy of 100%.

The action *Push* benefits the most from the action-dependent space-time modeling. As Figure 5-13 shows, this action is also characterized by strong localization variance. Preserving the action space-time layout implies a performance gain of 37% on UT1 and 23% on UT2.

5.7.4 Adaptive Grids: Unconstrained Data

The previous experiment focused on a data recorded in a controlled environment. We also want to assess the utility of the agBoW for "real world" video shot in unconstrained condition. We apply our approach on the UCF-Youtube dataset (see 2.3.3),

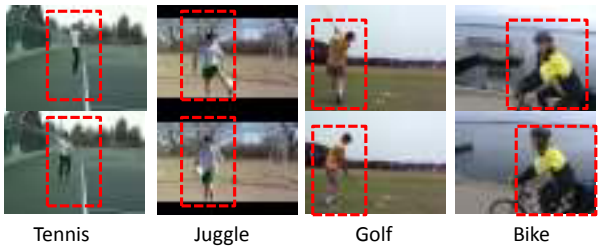


Figure 5-14: Youtube actions with static space-time context.

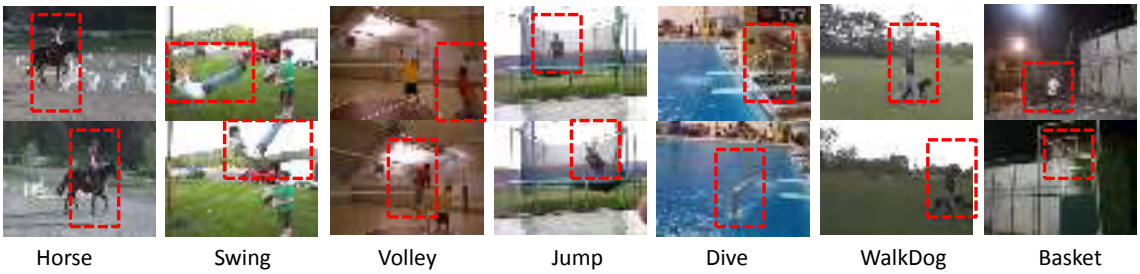


Figure 5-15: Youtube actions with dynamic space-time context.

containing 1668 videos extracted from the Youtube website, and the HMDB dataset (see 2.3.4), composed by 6849 online and movie videos.

UCF-YouTube

UCF-Youtube is composed by 11 actor-object actions which we divide in two categories as Figure 5-14 and 5-15 show. *Golf*, *Bike*, *Tennis*, *Juggle* are characterized by static space-time context, the action position remains stable over time, while *Basket*, *Dive*, *Swing*, *Jump*, *Volley*, *DogWalk*, *Horse*, have dynamic space-time context.

Table 5.3 reports the accuracies obtained on UCF-Youtube dataset. The agBoW representation outperforms BoW and spBoW on unconstrained videos as the result on UCF-Youtube shows. It reaches an average accuracy gain of 6.4% compared the BoW and 5.7% relatively to the spBoW. Task-specific space-time context modeling is therefore helpful for unconstrained video annotation.

In Table 5.4, we evaluate the agBoW impact for the dynamic and static space-

	Basket	Golf	Dive	Bike	Horse	Juggle
BoW	84.8	95.4	90.7	76.0	**95.5**	86.2
spBoW	82.7	98.5	86.1	**80.7**	91.0	87.9
agBoW	**90.2**	**94.9**	**93.6**	80.5	91.5	**88.2**
	Swing	Tennis	Jump	Volley	DogWalk	**Mean**
BoW	51.6	85.5	87.6	76.1	62.5	81.1
spBoW	51.9	88.0	**88.3**	78.1	62.2	81.6
agBoW	**72.1**	**88.2**	88.0	**88.4**	**73.4**	**86.3**

Table 5.3: Average Accuracies on the YouTube dataset.

Context Type	agBoW Accuracy	Gain compared to spBoW	Gain compared to BoW
Static	87.9	−4%	2.5%
Dynamic	85.3	12.6%	8.9%
Static + Dynamic	86.3	5.7%	6.4%

Table 5.4: Dynamic Spatial Context vs Static Spatial Context accuracy and performance gain on the YouTube dataset.

time context categories. For the static space-time context, agBoW does not lead to a significative improvement. It outperforms the BoW by 2.5% but sees its performances drop by 4% when compared to the spBoW. Since there is no strong action localization variation, a grid cell corresponding to an action region, is likely to capture only action statistic through time in the video. It will not mix with foreground and background information. In this case, the fixed grids are able to depict the actions space-time context with a sufficient precision in videos.

The agBoW representation does lead to a significative improvement of dynamic space-time context action. It obtains a performance gain of 8.9%, and 12.6% relatively to BoW and spBoW. Our task-specific context modeling is therefore especially useful for actions with a strong localization variation in time. For instance, the *Walk Dog* action generally involves a human moving from one extremity of the video to the other side. Due to their regular geometry, fixed grids will generally segment the action across different grid cells. By contrast, agBoW learns a segmentation grid with aperiodic geometry following the localization of an action through time. On

Walk Dog agBoW achieves a gain of 18% comparatively to the spBoW.

HMDB

We also evaluate agBoW on the challenging HMDB composed by 51 diverse actions which are extracted from movie and web videos.

Figure 5-16 compares the agBoW with a traditional BoW and spBoW representation. agBoW outperforms BoW of 12.5% relatively, from an average accuracy of 41.6% to 46.8%. It also outperforms by 5.6% the spBoW which achieves 44.3%. In addition, if we compare the classification scores per action with the fixed segmentation grid defined in 5-11, we observe that the Adaptive Grids obtain the best results in 32 out of 51 actions.

Adaptive Grids are therefore able to learn relevant information on the space-time shape of an action even with complex HMDB videos that are subject to strong camera motion and background clutter.

We also evalute the impact of the different Adaptive Grids parameters on the HMDB dataset in Figure 5-17.

Figure 5-17a alters the numbers of Adaptive Grids C learned for each action. It shows that increasing the number of grids improves the accuracy performances. But, increasing the grid number also leads to an augmentation of the action model dimensionality. Indeed, our action model $\mathbf{W} = [\mathbf{W}_1, ..., \mathbf{W}_C]$ have one group of coefficient $\mathbf{W}_i$ associated with each grid $\mathbf{g}_i$. Learning one more grid requires the addition of one group $\mathbf{W}_i$ in our model. In practice, we determine that $C = 16$ Adaptive Grids is a good trade-off between performance and efficiency.

Adaptive Grids are built in a bottom-up fashion by combining several cuboids which are defined through a regular grid. Figure 5-17b study the impact of the underlying grid layout. We observe that finer regular grids lead to better average accuracies. Optimally, we would like to consider directly the video voxels as cuboids to build our Adaptive Grids. However, an augmentation in the number of cuboids

composing the regular grids also increases the size of the generic intermediate representation. For instance, assuming we use a BoW vocabulary of 4096, a 4x4x4 regular grid already leads to generic intermediate representation with a dimensionality size of 262144. Considering a 5x5x5 grid would require the handling of a generic intermediate representation with a dimension of 512000. Practical limitations, such as computer RAM memory thus constraint the choice of the regular grid the layout of associated to Adaptive Grids.

Figure 5-17c evaluates the impact of γ weighting the grid regularizer importance $\Gamma(\mathbf{G}) = ||\mathbf{G} - \alpha\mathbf{S}||_2^2$. in our learning model (5.9), taking into account the structural information with the Γ regularizer in order to fit the local feature space-time distribution does help for classification.

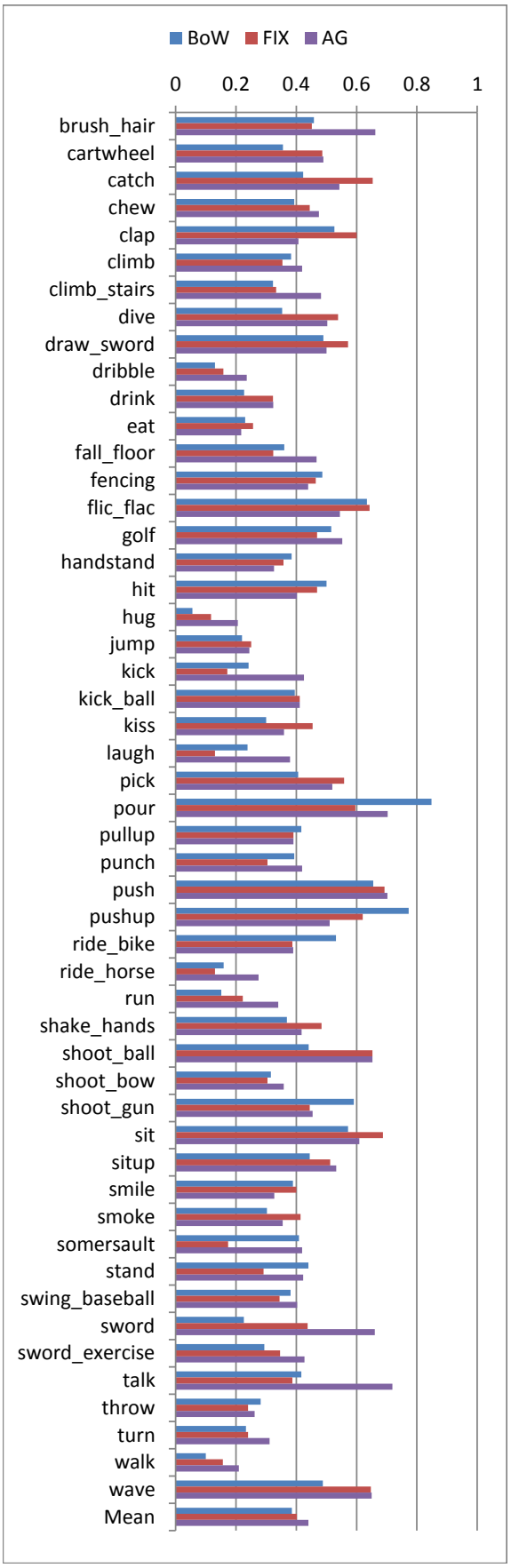
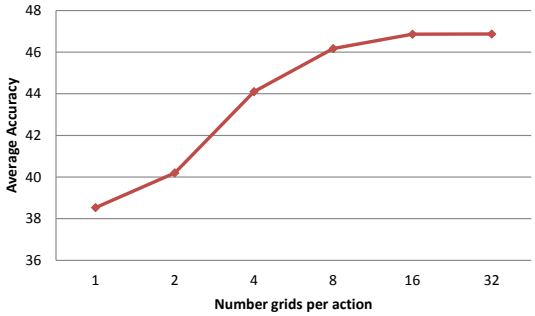
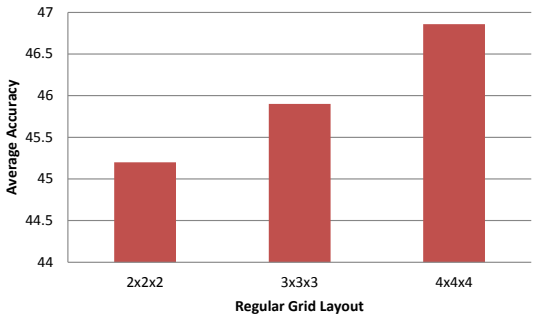


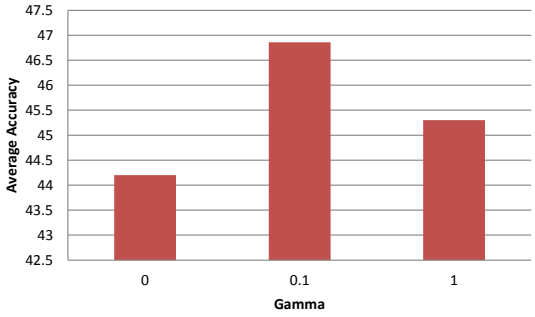
Figure 5-16: Per action Average Accuracy.



(a) Evaluation of learned grid number.



(b) Grids number



(c) Underlie grid layout

5.8 Conclusion

In this chapter, we introduced the Adaptive Grids which are a task-specific space-time context. Adaptive Grids aim at capturing the action-specific space-time information. They learn the action space-time shape by extracting both structural and appearance information from the training video data. We evaluate our proposal on 4 standard datasets. Our experimental study shows:

- space-time does indeed convey discriminative information. Video representations, which capture the space-time distribution of local features using either fix or adaptive segmentation grids, systematically outperform orderless-representation on action recognition;
- space-time context is action dependent, some grids are better fit to capture the action space-time contexts;
- by modeling action-dependent context, the adaptive grid leads to a performance gain of 5%, on average, compared to traditional approaches which rely on predefined and fix segmentations layout;
- our approach is especially efficient for actions which are subject to strong localization variations. In such case, Adaptive Grids, which are learned directly from the data, are able to follow the action throug time in the video. On the YouTube dataset, Adaptive Grids leads to a performance gain of 12.6% for such actions.

Chapter 6

Biological-Inspired Attention Context

This chapter introduces a biological-inspired attention context. It leads to a video representation that leverages the space and time context while remaining invariant to the global space-time transformations, like translation and rotation. Being robust to such transformations is of primary importance for unconstraint video where the action localizations can drastically shift between frames.

We evaluate our approach on the standard KTH, UCF50 and HMDB datasets. Our experimentation shows that biological-inspired attention context constantly outperforms other video space-time contexts by 10% on average.

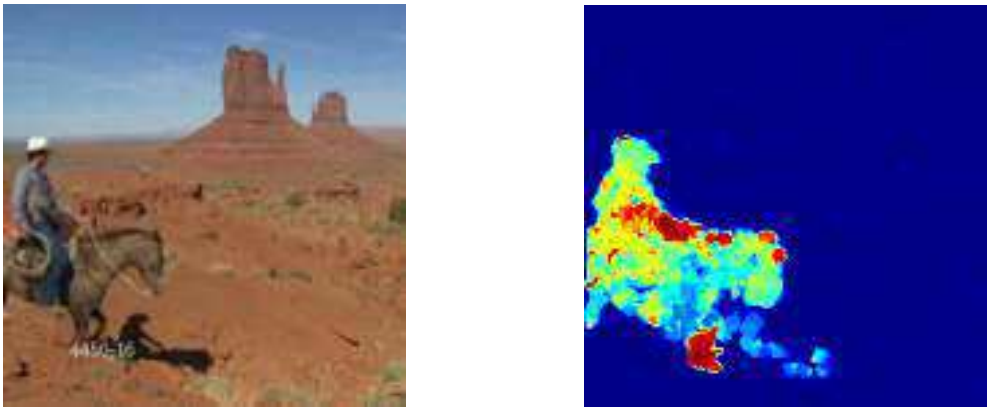
6.1 Motivation: Retaining the Space-Time Invariance

An abundant stream of data (around 10^9 bits) enters the human eyes every second [90]. To deal with this important amount of data, the human vision is able to identify regions of interest from an input visual content in a few milliseconds [185]. This mechanism, called visual attention, allows to restrict the visual analyze only to the most relevant visual input parts.

Definition 9. *Attention context: any information that measures the visual importance of video space-time regions; in term of how the regions appear to stand-out, for an observer, relatively to their immediate neighborhoods.*

In this chapter, we investigate the attention context from an action recognition perspective. Similarly to the space-time domain, we show that discriminative information is not equally distributed in visual attention domains. By taking into account video visual attention, we can augment our video representation and improve the action-recognition performance.

6.1.1 Visual Attention Context



Input frame Saliency map

Figure 6-1: Example of motion saliency map estimating the visual attention of an input video frame.

At core of visual attention lie an idea of selection and a notion of relevance. Visual attention indeed aims at highlighting informative parts of some visual content. Considering an image, an attention computational model produces a saliency map, *c.f.* Figure 6-1, which is a topographic map that depicts the image visual conspicuousness [21]. The main ideas behind the visual attention models date back to William James [71], who suggested that human select informative regions using bottom-up cues, extracted from the low-level visual data, and top-down task-dependent information.

Bottom-up approaches are driven by stimuli. They identify regions of interest based on their dissimilarity with respect to their immediate neighborhoods [67, 90, 185]. This dissimilarity is generally evaluated by a center-surround operation [67]. Given a local region and its immediate surroundings, both characterized by low-level local descriptors, the region saliency is defined as the divergence between the descriptors. Bottom-up saliency maps therefore exhibit the underlying structural organization of an image or a video. Different low-level visual descriptors can be used to characterize the local regions. It leads to attention models that focus on different structural characteristics (appearance, motion...) [21].

Top-down approaches use task-dependent information to select sub-regions in image or video [66]. As shown by the work of Yarbus [215], visual task at end governs the eye fixations for a human. Top-down approach models this psychological principle. They decide where to look depending on visual task that we want to accomplish.

In this chapter we do not propose a new visual attention computational model. We focus on how to exploit optimally existing visual attention model for action recognition.

6.1.2 Attention Context and Space-Time Information

Figure 6-2: Space-time context importance: "Soccer" and "Running" are likely to be distinguished by the area surrounding the human legs in the video lower part while "Clap" and "Wave" are more easily distinguished by the upper-bodies.

We saw in Chapter 5 that discriminative information is not equally distributed in the video space-time domain. As Figure 6-2 shows, action like *Soccer* and *Running*

are likely to be distinguished using the area surrounding the human legs in the video lower part while *Clap* and *Wave* are characterized by the upper-bodies. Modeling space-time context allows focusing on the most discriminative part of the video volumes. In addition, Chapter 5 has demonstrated that space-time information conveys discriminative information for action recognition.

Space-time context models, described in the literature [26, 56, 74, 102, 106, 166] and the one introduced in Chapter 5, work directly in the space-time domain. Most of them extend fix grid spatial pooling [102], they divide a video using space-time segmentation grids and pool the features locally in each grid cell.

Inter-Videos Intra-Video

Figure 6-3: Space-time variance: actions can be subject to localization variance due to camera viewpoint change in different videos. Even within a single video sequence, the action area is prone to change among frames.

Statement Despite performance improvement, those approaches lose the bag-of-words space-time invariance. Different action instances with various localizations in the space-time volume result in divergent representations. This problem is severe for actions that have dramatic space-time variance as illustrated in Figure 6-3. In this case, spatial pooling divides one action across different grid cells which may lead to a significant performance drop. A BoW representation robust to space-time variations is therefore critical for action recognition.

To overcome the action space-time variation, we propose to leverage the space-time information using saliency measures. We propose a new representation that takes advantage of the space-time discriminative context with an emphasis on retain-

ing the space-time robustness. Beyond standard spatial pooling which uses segmentation grids in the spatial domain, we identify regions of interest in a video through saliency. Our algorithm relies on the idea that the discriminative information has a non-uniform distribution in saliency spaces. For example, *Running* is more likely to be distinguished from *Walking* by regions with high salient motion. In addition, different saliencies highlight different regions in the video space-time volumes. They capture complementary information which can be appropriately exploited by the classifier.

6.1.3 Our Contribution

To sum-up, this chapter proposes two main contributions:

- We introduce a novel space-time invariant pooling which leverages the space-time context. We first extract video structural cues using various bottom-up saliencies. We then aggregate the local feature statistics over fixed saliency sub-regions, each sub-region defining a *structural primitive*. Focusing on different structural aspects, *cornerness*, *light* and *motion* saliencies are investigated. *Cornerness* highlights regions repeatable under geometric transformations, *motion* identifies regions with strong dynamics and *light* provides coarse object segmentation.
- We take advantage of WSVM to automatically determine the optimal *structural primitives* combination associated with a specific action. Each *structural primitive* corresponds to a particular space-time region. We want to learn which are the *cornerness*, *motion* and *light* subspaces that captures discriminative information for recognizing an action. Using a sparse feature weighting regularizer, we learn in a task-dependent and top-down fashion, what are the bottom-up saliency cues relevant to an action.

6.2 Related Work

This section provides a critical review of previous works related to our approach. Table 6.1 and 6.2 synthesize the differences.

Spatial pooling [102, 106] has successfully demonstrated performance improvement over classic BoW. However, to be fully effective, feature space-time statistics must align with the segmentation grids due to their fixed aspect ratio.

Recent efforts (cf Chapter 5) have tried to exploit richer spatial or temporal information by learning segmentation grids adapted to specific task. Jia [74] relies on sparsity to select segmentation grids in an overcomplete basis while Sharma and Harada [56, 166] learns segmentation weighted schemes. In Chapter 5 we also learn task-specific segmentation grids using video structural and discriminative appearance information. Since all those approaches partition local features in the spatial domain, they are not robust to space-time transformation. They remain sensitive to the action localization variability. In video, Cao [26] proposes a scene-adapted pooling. His approach focuses on modeling only the temporal context. Moreover, it is also not robust to time variation since the local features are divided in the temporal domain.

	Domain	Segmentation Adaptivity	Application
Lazebnik [106]	2D	Fix	Image
Laptev [102]	2D + time	Fix	Video
Harada [56]	2D	Class	Image
Jia [74]	2D	Class	Image
Sharma [166]	2D	Image	Image
Cao [26]	time	Video	Video
Adaptive Grids (Chap. 5)	2D+time	Class	Videos
Our approach	Saliency	Video	Video

Table 6.1: Comparison with other pooling methods taking into account the space-time context.

Saliency has already been used successfully in image analysis [132, 133, 143, 153, 164, 202]. Rahtu [153] uses saliency to segment object from image. In an image recognition context, Wang [202] rests upon bottom-up saliency to capture appearance information in low-level descriptors. Parikhn, Shabaz and Moosman [132, 133, 143, 164] define sparse sampling strategies to detect local features. Our motivation significantly differs from those approaches. We do not use saliency information to sample features

but to pool them [91]. We identify prominent regions in video using saliencies in order to model the space-time context while preserving the space-time robustness. In addition, our approach uses several saliency measures. It learns, with a sparse WSVM classifier, which are the discriminative saliency subspace associated to an action. Consequently, our approach model the space-time context using both bottom-up saliencies cues, and, task-dependent information learn from the WSVM classifier. While the works approaches were using either bottom-up or top-down information, we leverage both type of information.

	Saliency	Application
Rahtu [153]	Bottom-up	Segmentation
Wang [202]	Bottom-up	Descriptor Crafting
Sharma [166]	Top-Down	Spatial weighting
Mikolajczyk [132]	Bottom-up	Sampling
Moosman [133]	Top-Down	Sampling
Parikh [143]	Bottom-Up	Sampling
Shahbaz [164]	Top-Down	Sampling
Saliency Pooling	Top-Down + Bottom-up	Pooling

Table 6.2: Comparison with other works using visual attention.

6.3 Space-Time Robust Representation

In this section, we introduce a novel content-based pooling. This pooling operation captures the local features distribution with respect to the video structural information, extracted using saliency functions. We demonstrate that content-based pooling leads to a video representation which inherits from the space-time invariance properties of its corresponding saliency functions.

Figure 6-4 compares two pooling schemes which rely either on 2×2 static grid segmentation or on a motion saliency based dynamic segmentation. Due to its localization variance, the action falls in different cells of the static grids leading to two spatial BoWs having low-similarity despite depicting the same action. By segmenting the video dynamically, the second pooling scheme remains robust to the action space-

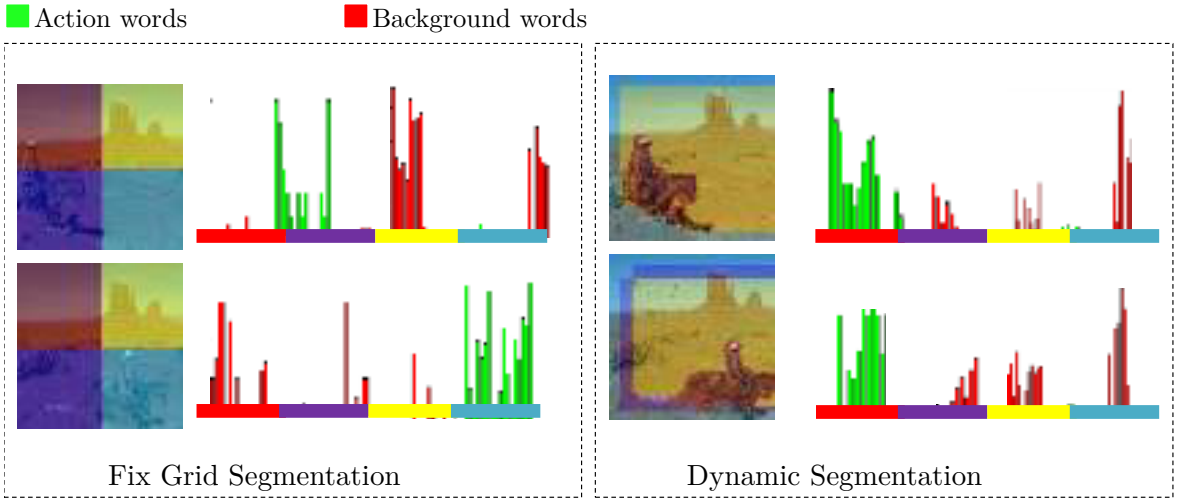


Figure 6-4: Space-time robustness importance. Due to the action shift, 2x2 grid results in spatial BoWs having a low similarity despite representing the same action. Pooling using dynamic segmentation remains robust to the action space-time variance while still modeling the feature space-time context.

time variance while still taking advantage of the local feature space-time context. This motivate us to propose a new pooling algorithm using video content information.

6.3.1 Content Driven Pooling

In the following, we first give another formulation of spatial pooling which is equivalent to the one described in Section 5.4, and then extend this formulation to take advantage of video saliency information.

Let $\mathbf{D} = \{\mathbf{d}_i\}_{i\in[1,M]}$ be a set of local features extracted from a video. We denote by $\mathbf{G} = \{\mathbf{G}^j\}_{j\in[1,C]}$ a set of grid cells. Each $\mathbf{G}^k$ is a binary matrix indicating which video voxels are active, $\mathbf{G}^j \in \{0,1\}^{s_x\times s_y\times s_t}$, (s_x, s_y, s_t) being the video dimension. Based on those definitions, we express the max spatial pooling operation as:

$$\mathbf{X}^j = \max_{(x,y,t)\in\mathbb{R}^{s_x s_y s_t}} \mathbf{G}^j_{(x,y,t)} \times code(\mathbf{d}_{\omega(x,y,t)}), \tag{6.1}$$

$\omega : \mathbb{R}^3 \rightarrow [1, M]$ is function indexing the descriptors $\mathbf{D}$ based on their positions. The function $code : \mathbf{D} \rightarrow \mathbb{R}^K$ is a local feature coding scheme. (6.1) is equivalent to the action-specific pooling introduced in section 5.4, but, we index the grid us-

ing directly the voxels position rather than grid cuboids. Indexing directly the grid based on the video voxels allows us to emphasize the connection between spatial pooling and content driven pooling. Indeed, traditional spatial pooling uses a set of pre-defined pyramidal grids segmenting the video in increasingly finer cells. Recent pooling works [56, 74, 166] and the approach developed in Chapter 5 learns $\mathbf{G}$ directly from data achieving task-specific segmentation. They all focus on modifying $\mathbf{G}$ in (6.1) to obtain a better segmentation scheme. While increasing the flexibility of spatial pooling, these methods still result in space time division and are unable to handle the video-specific space-time variance.

Differently, we aim at modeling the space-time context while remaining robust to the space-time variance.

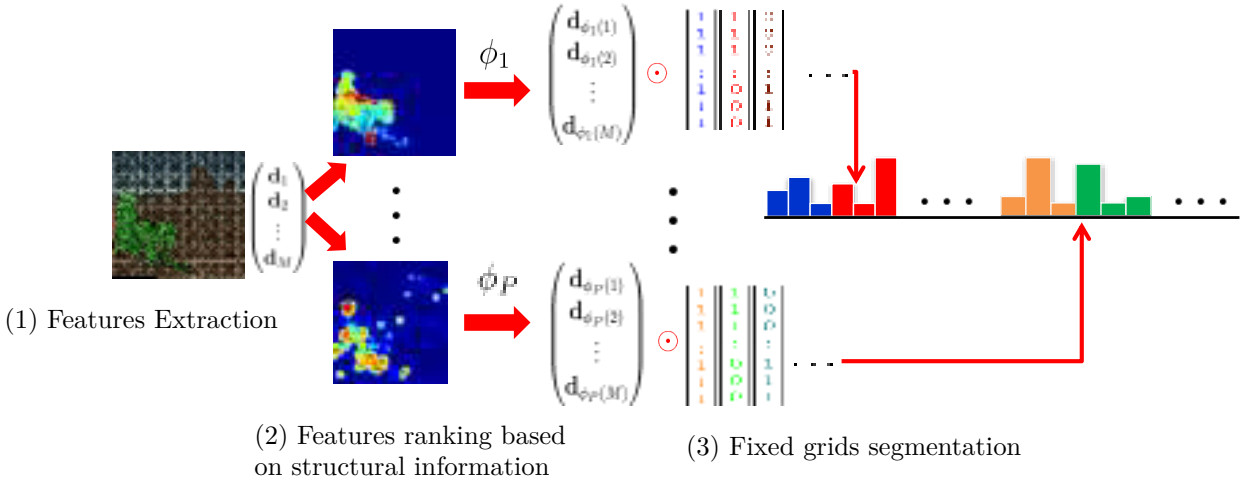


Figure 6-5: Space-Time Invariant Pooling. By segmenting in the saliency space, accordingly to their sailent rank, our representation remains invariant to global space-time transforms.

To do so, we identify prominent regions using saliency. As shown in Figure 6-5, we (1) extract saliency information from a video; (2) order local features in rank lists according to each saliency; (3) capture local feature statistics in various rank list sub-regions. As a result, our pooling scheme does not require the local feature absolute space-time coordinates to capture the video space-time context. Indeed, saliencies are computed using the feature relative-positions. Content-based pooling performs

video-specific segmentation based on their structural cues. In addition, since we use saliency ranks to group features instead of their absolute saliency values, our representation remains invariant to global translation in the saliency space.

To formulate our content driven pooling, we modify the indexing function ω in (6.1) to include video saliency cues. Let $\mathbf{P} = \{p_i\}_{i\in[1,M]}$ be the saliency values for each local feature. We introduce $\phi : [1, M] \rightarrow [1, M]$, a bijective function that orders the local features according to $\mathbf{P}$. To infer $\Phi = \{\phi(i)\}_{i\in[1,M]}$, we minimize the functional

$$\hat{\Phi} = \min_{\Phi\in\{[1,M]\rightarrow[1,M]\}} \sum_{i=1}^{M} i p_{\phi(i)}. \tag{6.2}$$

$\hat{\Phi}$ solving (6.2), $d_{\hat{\phi}(1)}$ is the local features having the highest saliency while $d_{\hat{\phi}(M)}$ corresponds to the lowest one. Relying on $\hat{\Phi}$ definition, we can now introduce the content-based pooling operation:

$$\mathbf{X}_{i,k} = \max_{j\in[1,M]} \mathbf{G}_j^{i,k} \times code(\mathbf{d}_{\hat{\phi}(j)}). \tag{6.3}$$

With (6.3), the pooling is performed in the saliency domain instead of the space-time domain. $\mathbf{G}$ is defined as a one dimensional pyramidal tiling. We consider sequence of segmentation grids $\mathbf{S}^0 \dots \mathbf{S}^{L-1}$ such as each grid $\mathbf{S}^i$ is composed by 2^i equally sized cells: $\mathbf{G} = \{\mathbf{G}^{i,1}, ..., \mathbf{G}^{i,2^i}\}$ where

$$\mathbf{G}^{i,k} \in \{0,1\}^M \text{ with } \mathbf{G}_l^{i,k} = \begin{cases} 1 & \text{if } l \in [\frac{k-1}{2^i}M, \frac{k}{2^i}M] \\ 0 & \text{Otherwise} \end{cases} \tag{6.4}$$

Definition 10. *Structural Primitive: Visual signature characterizing a saliency sub-space of an image or a video.*

$\mathbf{X}_{i,k}$ captures the distribution of local features over a saliency sub-region. It defines a *structural primitive*. The *structural primitives* are $\|.\|_2$ normalized and concatenated

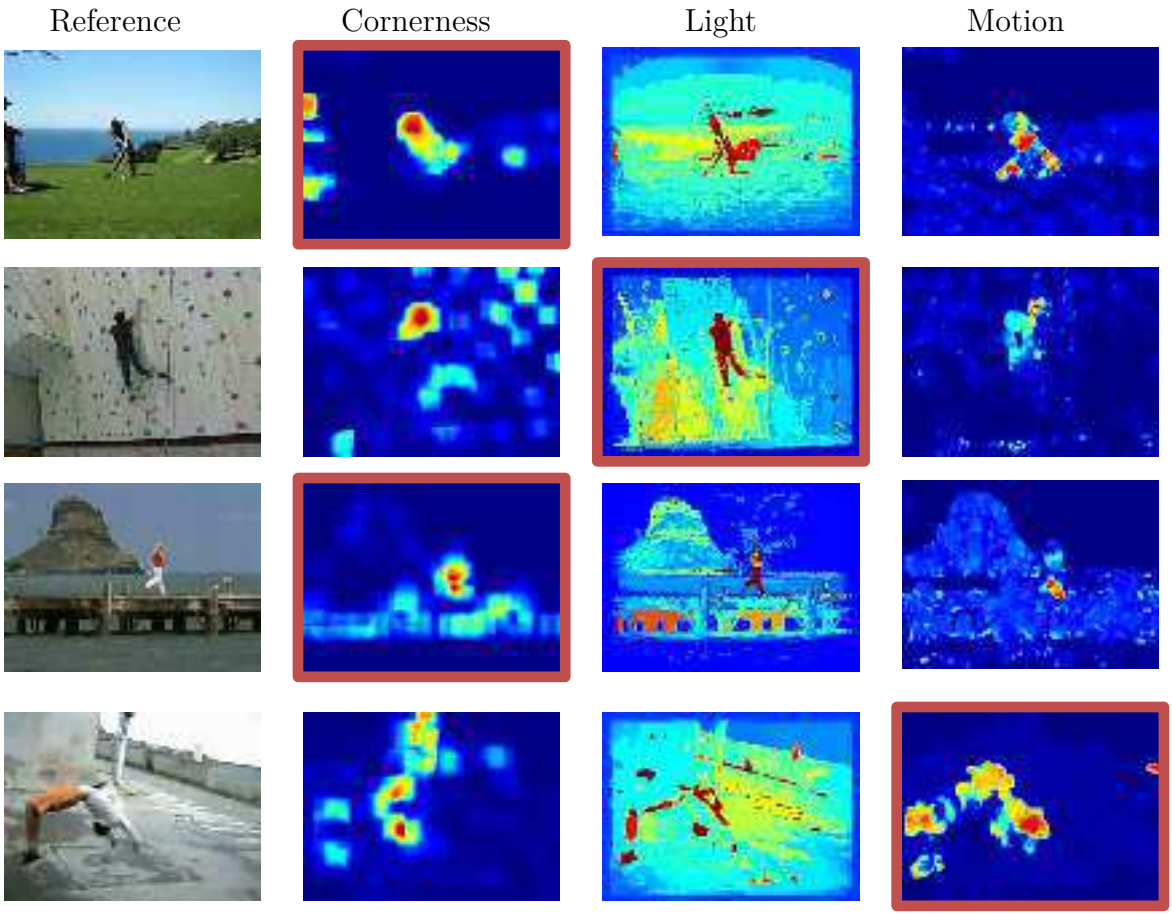


Figure 6-6: Prominent areas highlighted by the different saliency measures. Red contour indicates which saliency function obtain the best overlap with the actual action localization.

to obtain the signature $\mathbf{X} = [\mathbf{X}_{1,1}, ..., \mathbf{X}_{L-1,2^{L-1}}]$. When using several saliency functions, we repeat this pooling operation for each measure.

6.3.2 Saliency Measures

To complete our content driven pooling formulation, the values $\mathbf{P} = \{p_i\}_{i\in[1,M]}$ need to be defined. $\mathbf{P}$ values take advantage of the video structural cues through saliency measures.Let $s : \mathbf{D} \rightarrow [0-1]$ be a saliency funcion. We define $p_i \in \mathbf{P}$ as:

$$\forall i \; p_i = s(\mathbf{d}_i). \tag{6.5}$$

s is a local measure which describes how much a feature differs relatively to its immediate neighborhoods [67]. (6.5) slightly abuses the $\mathbf{d}_i$ notation: when used in the context of the saliency function s, $\mathbf{d}_i$ doesn't designate a local feature descriptor but a local feature localization, $\mathbf{d}_i = (x_i, y_i, t_i, s_i)^T$, s_i being the feature scale value.

In practice, we focus on 3 different saliency functions highlighting the "cornerness", "light" and "motion" structure of video. The cornerness saliency selects visually distinctive regions, which are repeatable under geometric transformations. Region cornerness is estimated with the Harris-Laplace transform [132]. Differently, light and motion saliency use an efficient sliding windows based center-surround operation [153]. Light provides coarse object segmentation. Motion saliency considers the video optical flow computed for each video frame through the Farneback algorithm [41]. We detail the different saliency functions in the following.

Cornerness The cornerness saliency determines the visual distinctiveness associated to local features $\mathbf{d}_i$. Feature cornerness is estimated using the video frame gray values with the Harris-Laplace function [132] which is based on the second moment matrix of the normalized derivatives (Laplacian). This matrix, also called the auto-correlation matrix is used to highlight local regions having an affine shape:

$$\mu(\mathbf{d}_i)) = s^2 g(\sigma_d) \begin{bmatrix} L_x^2(\mathbf{d}_i) & L_x L_y(\mathbf{d}_i) \\ L_x L_y(\mathbf{d}_i) & L_y^2(\mathbf{d}_i) \end{bmatrix},$$

where $L_x(\mathbf{d}_i)$ (respectivelly $L_y(\mathbf{d}_i)$) is the gray-level intensity derivative computed according to x (respectivelly y) at the feature $\mathbf{d}_i$ position and scale. $g(\sigma_d)$ is a Gaussian smoothing. This matrix eigen values describe the principal signal changes in the neighborhood of p [132]. Using this property, we can design a saliency measure (eq 6.6) having high value for point where both curvatures in x and y-dimensions are significant:

$$s(\mathbf{d}_i) = \frac{\rho_{min}(\mu(\mathbf{d}_i))}{\rho_{max}(\mu(\mathbf{d}_i))}. \tag{6.6}$$

In (6.6), $\rho_{min}(\mu(\mathbf{d}_i))$ and $\rho_{max}(\mu(\mathbf{d}_i))$ corresponds respectively to the minimum and

maximum eigen value of $\mu(\mathbf{d}_i)$. Cornerness property prioritizes local features which are repeatable under geometric transformations: it highlights relatively small ellipsoidal object such as mouth or nose in face close-up view.

Illumination The illumination saliency emphasizes regions with homogenous reflectance. Each video RGB frame is transformed into the CIELab color space [39]. The L (Light) component of the color space is divided in 60 equal-sized bins and the light saliency is computed by a center-surround operation using the bins distribution contrast between sliding windows inner and outer regions [153].

The center-surround operation considers a rectangular window W divided into two disjoint parts, a rectangular inner window K (kernel) and the outer windows B (border). It applies the hypothesis that points in K are part of the foreground and points in B are part of the background. We denote by $\mathbf{K} \in \mathbb{R}^{1\times 60}$, respectively $\mathbf{B} \in \mathbb{R}^{1\times 60}$, the quantized light histogram of the inner windows K, respectively the border B, of the W windows centered on the feature $\mathbf{d}_i$ localization. $\mathbf{K}$ and $\mathbf{B}$ estimate the light distribution in the inner and outer windows. They are convolved with a Gaussian to increase their robustness toward quantization error [153]. The saliency of region $\mathbf{d}_i$ is equal to the light distribution divergence between K and B:

$$s(\mathbf{d}_i) = \frac{p\mathbf{K}_{q(\mathbf{d}_i)}}{p\mathbf{K}_{q(\mathbf{d}_i)} + (1-p)\mathbf{W}_{q(\mathbf{d}_i)}}, \tag{6.7}$$

$q(\mathbf{d}_i)$ is the L-component quantized value at the position $\mathbf{d}_i$ and p is a prior indicating the likelihood of $\mathbf{d}_i$ being part of the foreground. In practice, we set p to 0.2 following [153].

To achieve robustness toward scale variation, we use several sliding windows of different size. Let $\{W^w\}_{w\in[1,4]}$ be 4 sliding windows with a row and column size equal to $(25, 10)$, $(30, 30)$, $(50, 50)$ $(70, 40)$ percents of the video frame dimension, and, $\{\mathbf{K}^w\}_{w\in[1,4]}$ their corresponding inner and outer histograms. The light saliency becomes:

$$s(\mathbf{d}_i) = \max_{w\in[1,4]} \frac{p\mathbf{K}^w_{q(\mathbf{d}_i)}}{p\mathbf{K}^w_{q(\mathbf{d}_i)} + (1-p)\mathbf{W}^w_{q(\mathbf{d}_i)}}, \tag{6.8}$$

A region with a homogenous light reflectance is likely to correspond to an object part such as a human body part.

Motion The last saliency segments videos according to motion information [110]. This motion saliency considers the video optical flow computed for each video frame through the Farneback algorithm [41]. The flow magnitude is quantized into 16 uniform bins. The corresponding saliency is then computed with the same sliding windows approach than for light saliency [153], see (6.8). It results in a measure highlighting local feature with homogenous motion that differs from its neighborhood. Such features can be particularly discriminative since we want to characterize dynamic actions. However, this saliency suffers from camera motions and background dynamic clutter.

6.3.3 Space-Time Invariance Property

Content-based pooling leverage the space-time context of a video by emphasizing specific regions using saliency cues. In addition, content-based pooling inherits from the space-time invariance property of the salience function.

Theorem 2. *Given a saliency function s which is invariant to the translation, rotation and scale transformations, its corresponding content-driven pooling representation remains stable under those transformations.*

Proof. Let consider a saliency function $s : \mathbf{D} \rightarrow [0-1]$ that is invariant to translation, rotation and scale transformations:

$$s(\mathbf{R}\mathbf{d}_i + \mathbf{v}) = s(\mathbf{d}_i) \tag{6.9}$$

In (6.9), $\mathbf{R} \in \mathbb{R}^{3\times3}$ is an affine transformation matrix representing a scaling and a rotation, $\mathbf{v} \in \mathbb{R}^3$ is a translation vector. We also consider a set of local features $\mathbf{D} = \{\mathbf{d}_i\}_{i\in[1,M]}$, their corresponding saliency values $\mathbf{P} = \{p_i\}_{i\in[1,M]}$ and their ranking order $\mathbf{\Phi} = \{\phi_i\}_{i\in[1,M]}$ defined as in subsection 6.3.1.

To show the invariance property, we apply a global transformation to the video local features:

$$\mathbf{D}' = \{\mathbf{R}\mathbf{d}_i + \mathbf{v}\}_{i\in[1,M]}. \tag{6.10}$$

Due to the invariance property of s (6.9), the saliency values $\mathbf{P}' = \{p'_i\}_{i\in[1,M]}$ of the local descriptor $\mathbf{D}'$ remain stable under this transformation:

$$\forall i\ p'_i = s(\mathbf{d}'_i) = s(\mathbf{R}\mathbf{d}_i + \mathbf{v}) = s(\mathbf{d}_i) = p_i. \tag{6.11}$$

Consequently, the index set $\mathbf{\Phi}' = \{\phi'_i\}_{i\in[1,M]}$, ranking the $\mathbf{D}'$ saliency values through the minimization of

$$\hat{\Phi}' = \min_{\Phi'\in\{[1,M]\to[1,M]\}} \sum_{i=1}^{M} i p'_{\phi'(i)}, \tag{6.12}$$

also remains unchanged by the geometric transformations. Hence, content driven pooling remains stable under the translation, rotation and scale transformations. Indeed, considering a segmentation grid $\mathbf{G} \in \mathbf{R}^M$, the content driven pooling of $\mathbf{D}$ is equal to the content driven pooling of $\mathbf{D}'$:

$$\max_{j\in[1,M]} \mathbf{G}_j \times code(\mathbf{d}_{\phi(j)}) = \max_{j\in[1,M]} \mathbf{G}_j \times code(\mathbf{d}'_{\phi'(j)}). \tag{6.13}$$

□

This section shows that the space-time robustness of the attention based representation lies upon the space-time invariance property of the saliency functions. One can easily choose the video signature invariance level which fits its need by carefully selecting its representation saliency functions. In our case, cornerness is invariant to scale and affine transformations [132]. Experimental studies show that motion and light saliencies remain robust under affine transformations, however the formal invariance has not been demonstrated [153].

6.4 Top-Down Weighting

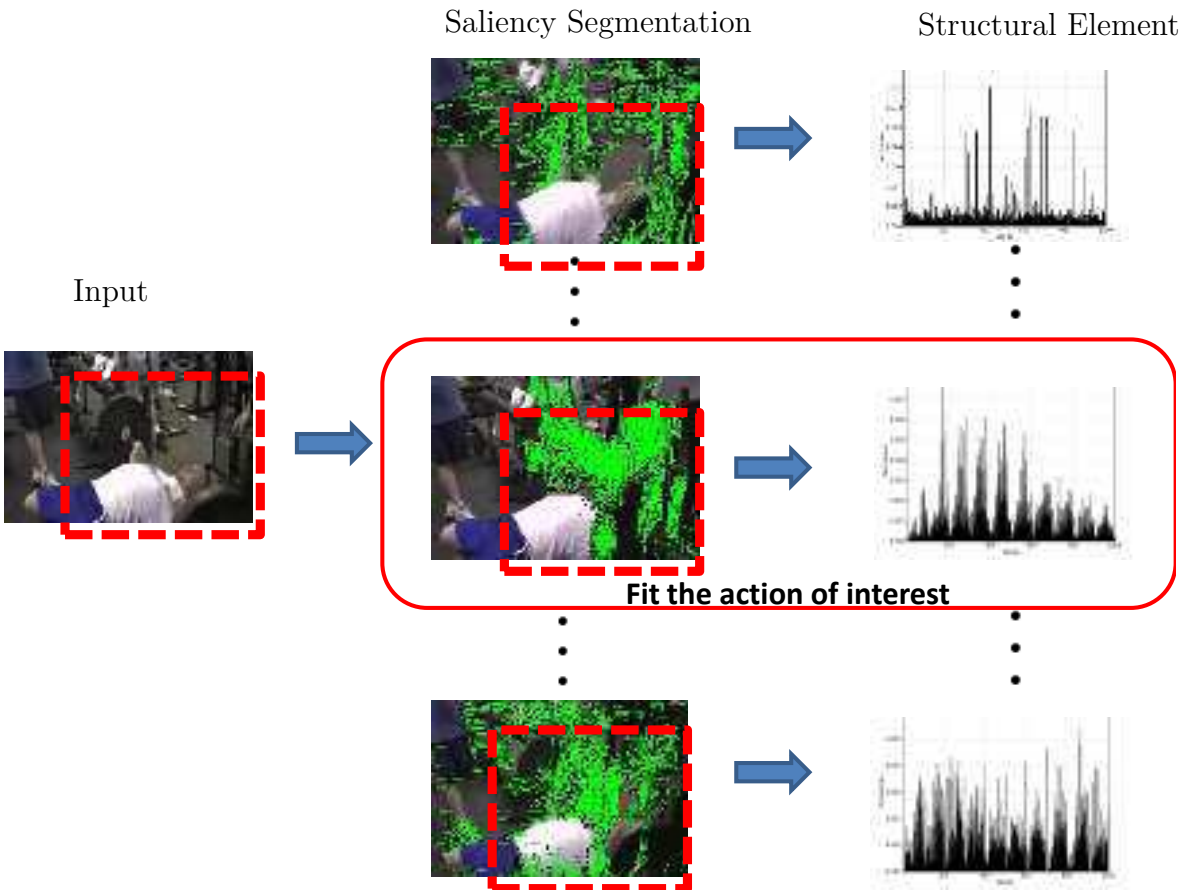


Figure 6-7: Different *structural primitives* highlights difference space-time region in the video. Using top-down information, we want to select the region that fit the action.

Content driven pooling results in a set of *structural primitives*, Each *structural primitive* characterizes some space-time regions of the video volume which correspond to a particular saliency subspace. As illustrated in Figure 6-6, saliency measures, therefore their *structural primitives*, emphasize different areas of the video space-time volume. Figure 6-7 shows that some *structural primitives* will capture information mainly extracted from the video background while other primitive will be more focused on the action foreground. Consequently, the *structural primitive* discriminative power is non-uniform, *i.e.* there are not equally discriminative for an action.

To leverage the non-uniform discriminative power of the *structural primitives*,

we propose to embed sparsity in the *structural primitives* selection. By focusing on only a few *structural primitives* at classification, we could take advantage of saliency functions which fit best the action of interest while discarding the one containing irrelevant or noisy information. We take advantage of the WSVM model to perform this selection (see Figure 6-8).

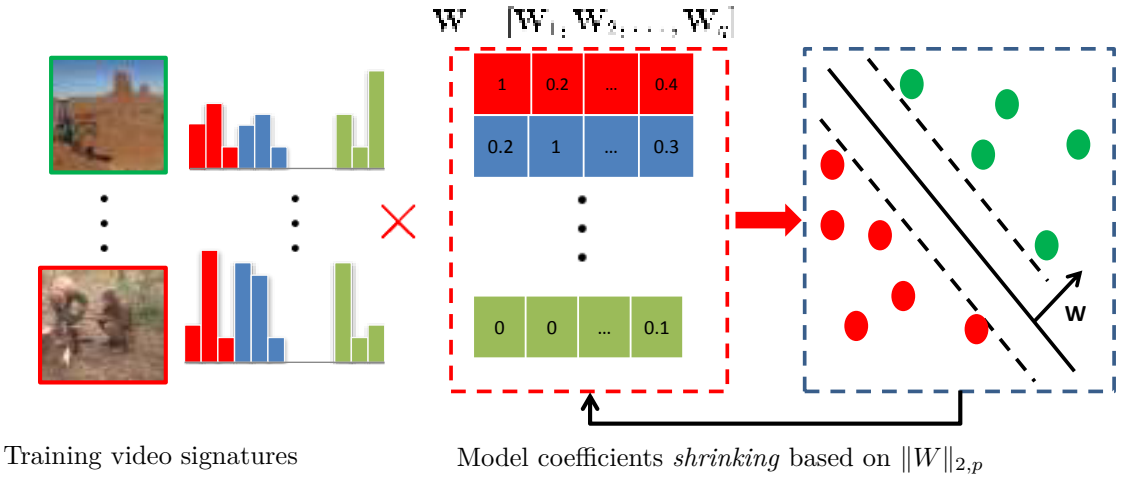


Figure 6-8: Attention Context Combination.

Let $\mathbf{X} \in \mathbb{R}^{N \times d}$ be N training video signatures and $\mathbf{Y} \in \{0,1\}^N$ their corresponding binary labels. Each video signature $\mathbf{X}_i$ is the concatenation of C *structural primitives i.e.*, $\mathbf{X}_i = [\mathbf{X}_{i,c}]_{i \in [1,C]}$. A *structural primitive* captures the local feature distribution over one subregion of one saliency space. We consider a linear model $\mathbf{W} = [\mathbf{W}_i]_{i \in [1,C]} \in \mathbb{R}^d$ with its bias term $b \in \mathbb{R}$. $\mathbf{W}_c$ is the group of $\mathbf{W}$ coefficients correlating with the *structural primitive* $\mathbf{X}_{i,c}$. In other words, $\forall i$ $\mathbf{X}_{i,c}$ defines one context associated with one saliency subspace. $\mathbf{W}_c$ is its corresponding model.

To induce sparsity in the *structural primitive* selection, we consider each *structural primitive* as a particular video context. Applying the WSVM defined in section 3.4, our model becomes:

$$E(\mathbf{W}, b) = \sum_{i=1}^{N} L(\mathbf{Y}_i, \sum_{c=1}^{C} \mathbf{X}_{i,c}\mathbf{W}_c + b) + \lambda\Omega(W), \tag{6.14}$$

where L is the square hinge loss:

$$L(\mathbf{Y}_i, \sum_{c=1}^{C} \mathbf{X}_{i,c}\mathbf{W}_c + b) = \max(0, \mathbf{Y}_i(\sum_{c=1}^{C} \mathbf{X}_{i,c}\mathbf{W}_c + b))^2, \tag{6.15}$$

and Ω is the regularizing term. As stated in section 3.4, the SVM model uses a $\|.\|_2$ norm as regularizer [210]. This norm attaches the same importance to each coefficient in $\mathbf{W}$, *i.e.*, each group $\mathbf{W}_c$ contributes equally. To leverage the non-uniform discriminative power of *structural primitives*, WSVM proposes to prioritize only the most substantial groups $\mathbf{W}_c$ for an action while discarding the irrelevant one by adding a sparsity constraint on $\mathbf{W}$.

Sparsity is induced through the use of a $\|.\|_{2,p}$ norm with $p < 2$. It uses a $\|.\|_{2,p}$ norm, a combination of a $\|.\|_p$ norm at the groups level and a $\|.\|_2$ norm at the individual coefficient level. While selecting only a few groups with the $\|.\|_p$ norm, $\|.\|_{2,p}$ considers the coefficient inside a group as a whole through the $\|.\|_2$, taking advantage of their implicit relation. Using the sparse regularizer, our model becomes:

$$E(\mathbf{W}, b) = \sum_{i=1}^{N} L(\mathbf{Y}_i, \mathbf{X}_i\mathbf{W} + b) + \lambda||\mathbf{W}||_{2,p} \tag{6.16}$$

WSVM learns, accordingly to the training dataset $(\mathbf{X}, \mathbf{Y})$, which are the *structural primitives* relevant for an action. For each specific action, it learns which are the saliency bottom-up saliency subspaces that contain discrimininative information. In this the sense, the WSVM correspond to the definition of a top-down saliency.

6.5 Attention Context Performances: Evaluation

In this section we evaluate the performance of the content based pooling and WSVM model on three action datasets: KTH, UCF50, and HMDB (see Section 2.3).

6.5.1 Experimental Setting

We compare our content-based pooling approach (structural-BoW) with a BoW representation, and, a fix grid spatial pooling based bag-of-words (spBoW). BoW discards the space-time context information while spBoW embeds space-time information relying on fixed and predefined segmentation grids. BoW is constructed using dense trajectories, LLC coding and max-pooling. Since a trajectory spans on several video frames, the average saliency value of its points defines the saliency value associated to the feature. For space-time invariant pooling, we segment each saliency space with 1, 2 and 3 cells segmentation grids leading to a total of 7 BoWs. We compare our approach with spatial pooling using a 2x2x2 segmentation grid. When they are not specified, the WSVM parameters are set as $\lambda = 0.1$ and $p = 1.5$. Those values have empirically demonstrated robust performances across the different datasets.

6.5.2 When Do Saliency Cues Help for Action Recognition?

	KTH	UCF50 5 folds	UCF50 25 folds	HMDB
BoW	93.7	86.7	85.3	41.6
Co	94.0	88.0	87.3	42.8
Li	93.8	90.2	89.6	42.5
Mo	**94.2**	**90.8**	**89.7**	**43.5**

Table 6.3: Average accuracies of BoW, structural-BoWs. Mo, Li and Co correspond respectively to Motion, Light and Cornerness structural-BoWs.

In a first experiment, we compare each individual structural-BoW using only one saliency function to the traditional BoW [170].

Results are reported Table 6.4. It shows that that cornerness, motion and light structural-BoWs always outperforms BoW. By taking into account the distribution of local features in the saliency domains, we obtain a performance gain up to 4.5%. Discriminative information is therefore not uniformly distributed in the saliency spaces.

Figure 6-9 verifies that the performance gain is not due to the signature dimensionality increase. Indeed, due to the 1,2 and 3 cells saliency segmentation, structural-

BoWs have the same dimension than 7 orderless BoWs. By increasing the BoW signature size, we don't reach the structural-BoW accuracies. Therefore structural-BoW performances improvement is not solely due to their size

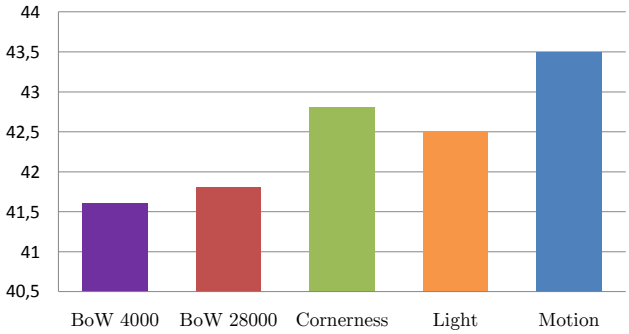


Figure 6-9: Impact of the vocabulary dimension. Average accuracy is reported.

Table 6.4 shows that motion has on average the best performance compared to the other structural-BoWs. However, if we consider the accuracy per action, illustrated in Figure 6-11, we actually observe that the different saliencies are complementary.

For example, on HMDB, cornerness obtains the best performances for the actions *Smile*, *Smoke*, *Eat*. As described by Figure 6-12, those actions are characterized by close-up face views. Cornerness focuses on visually distinctive local features. In this case, it highlights features located around the noise, eye or mouth area (Figure 6-10). Cornerness is also useful for actions such as *Catch*, *Golf* involving objects with relatively small ellipsoidal shape.

Light gets the best performances for the actions *Climb*, *Fall Floor* or *Shooting Bow* where an upper human body is present [95]. Light saliency performs a coarse segmentation which groups together the features associated to the human body in those actions (Figure 6-10).

Motion achieves the best performance on actions which are characterized by a strong motion (*Chew*, *Run*, *Flic Flac*...) where the local features having high motion saliency values are likely to be part of the action of interest (Figure 6-10).

More generally, a structural-BoW achieves significant performance improvement over a representation ignoring the space-time context when the pooling of the high

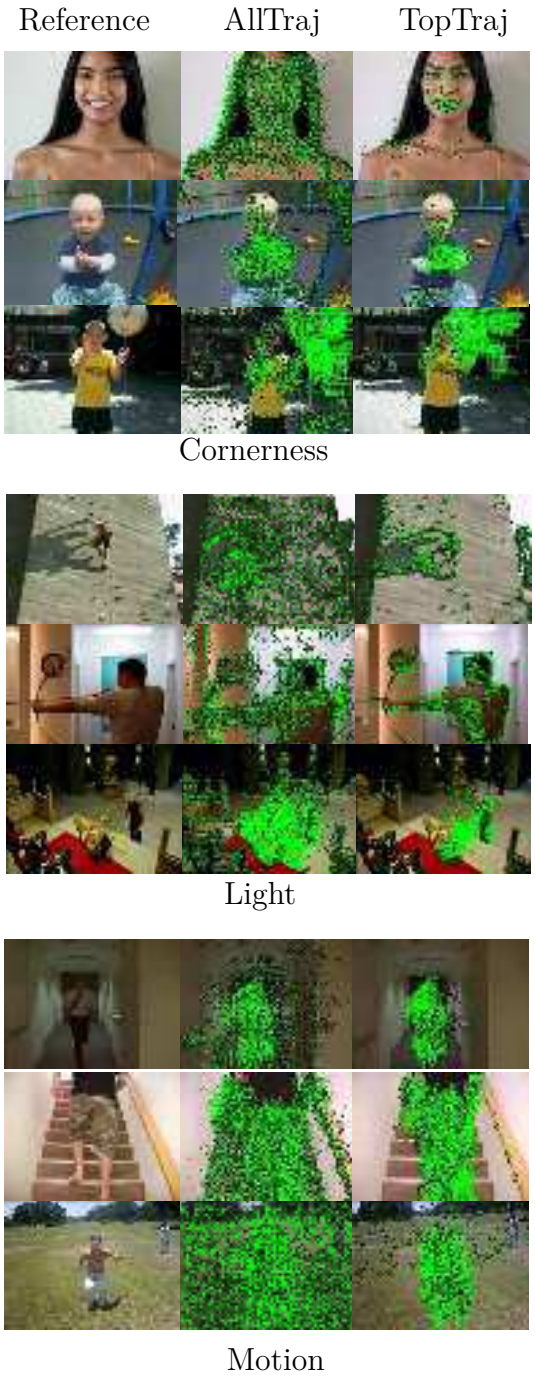


Figure 6-10: Prominent $\mathbf{W}_g$ groups in $\mathbf{W}$. The left column contains the reference frames. The middle column shows the extracted trajectories. The right column represents only the trajectories associated to the action most relevant *structural primitive*, i.e., the trajectories associated with the group $\mathbf{W}_g$ having the highest $\|.\|_2$ norm in $\mathbf{W}$. The most relevant *structural primitive* can be computed using cornerness, motion or light saliency depending on the action.

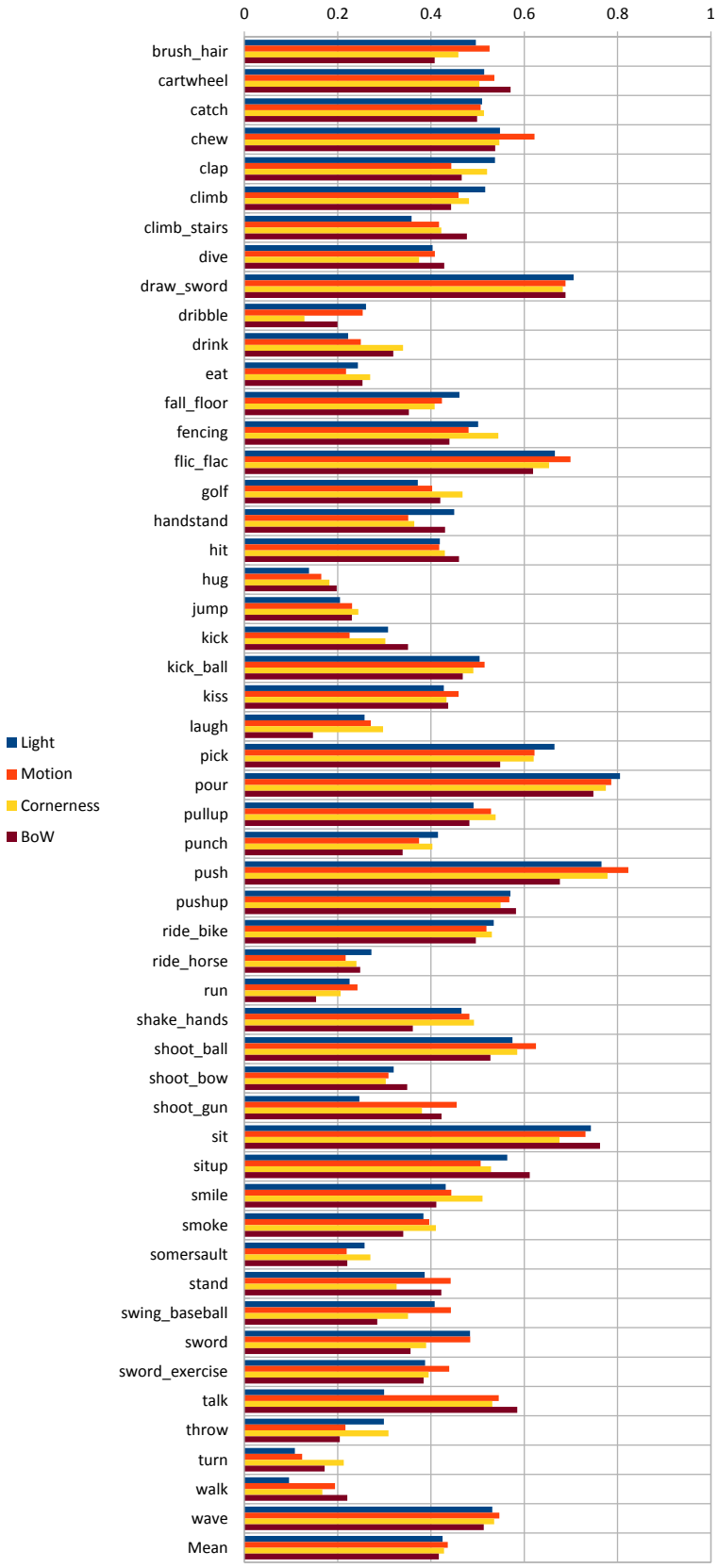


Figure 6-11: Per action average accuracy on HMDB.

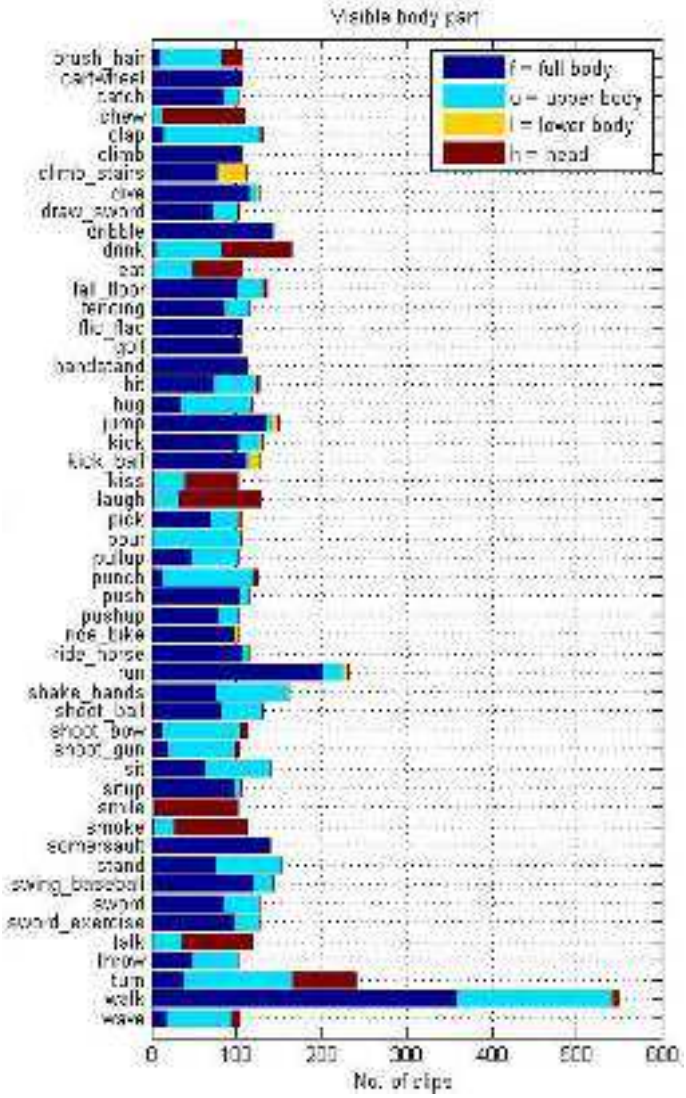


Figure 6-12: Action Properties (courtesy of Kuehne [95]).

saliency features only reduces the impact of the background clutter and leads to a more discriminative signature.

6.5.3 Are the Saliencies Complementary?

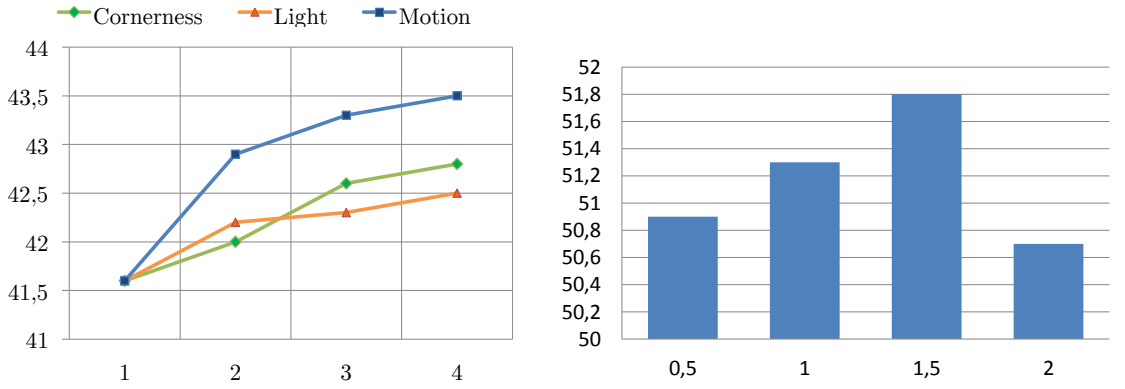
In this second experiment, we evaluate the combination of the different structural-BoWs through the WSVM.

	KTH	UCF50 5 folds	UCF50 25 folds	HMDB
spBoW	94.0	91.2	89.3	44.0
Mo + Li	94.2	91.7	90.6	45.9
Mo + Li + Co	94.4	92.5	91.3	48.5
Mo + Li + Co + spBoW	**94.6**	**94.1**	**92.8**	**51.8**

Table 6.4: Average accuracies of structural-BoWs, Spatial-BoW and their combinations. Mo, Li, Co and spBoW correspond respectively to Motion, Light, Cornerness and Spatial BoWs.

Table 6.4 reports the average accuracies of the spatial BoW and the structural-BoWs combination. On the HMDB dataset, a performance gain of more than 11%, from 43.5 to 48.5, is achieved by the structural-BoW combination (Co+Li+Mo) compared to the best individual structural-BoW (Mo). Table 6.4 therefore shows the complementarities of saliency based representations. Furthermore, by adding spatial BoW to our video signature, another improvement of 7% is obtained. Hence, spatial and structural-BoWs capture complementary information. The same trend can be observed on the UCF50 dataset. In the 25 fold setting, the combination of structural-BoWs achieves an average accuracy of 91.3 compared to 89.7 for Mo. By adding spatial information, we reach 92.8.

On the KTH dataset, structural-BoWs as well as their combination only slightly improve over the traditional and spatial BoW. Structural-BoWs combination achieves



(a) Impact of pyramid level number

(b) Impact of p on structural-BoWs combination

Figure 6-13: Impact of different parameters on the HMDB dateset. Average accuracy is reported.

a performance of 94.6 compared to 93.7 for a traditional BoW. KTH videos have almost static videos with no clutter. Most of the extracted features correspond to the foreground action, i.e. most of them are relevant to the action. It limits the need of modeling the space-time context. It should be noticed that spatial-BoW provides also a very limited improvement on this dataset, 94.0 against to 93.6.

Finally, as Table 6.4 shows, structural-BoW combination (Co+Li+Mo) always outperforms the spatial-BoW for each dataset showing the importance of space-time robustness. Based on WSVM, we represent visually the trajectory features corresponding to the $\mathbf{W}_g$ having the most impact for specific actions in Figure 6-10.

6.5.4 Parameters Evaluation

Regarding the parameters, Figure 6-13a evaluates the influence of the pyramidal tiling level number on the HMDB dataset. Adding more levels increase the performance up to a certain point for each saliency. To limit the dimension of our signature, we use 3 pyramidal levels that divide recursively the saliency space in one, two and three uniform bins.

Figure 6-13b evaluates the impact of the sparsity parameter p on the HMDB dataset. When $p = 1.5$, WSVM outperforms a standard SVM ($p = 2$) from 50.7

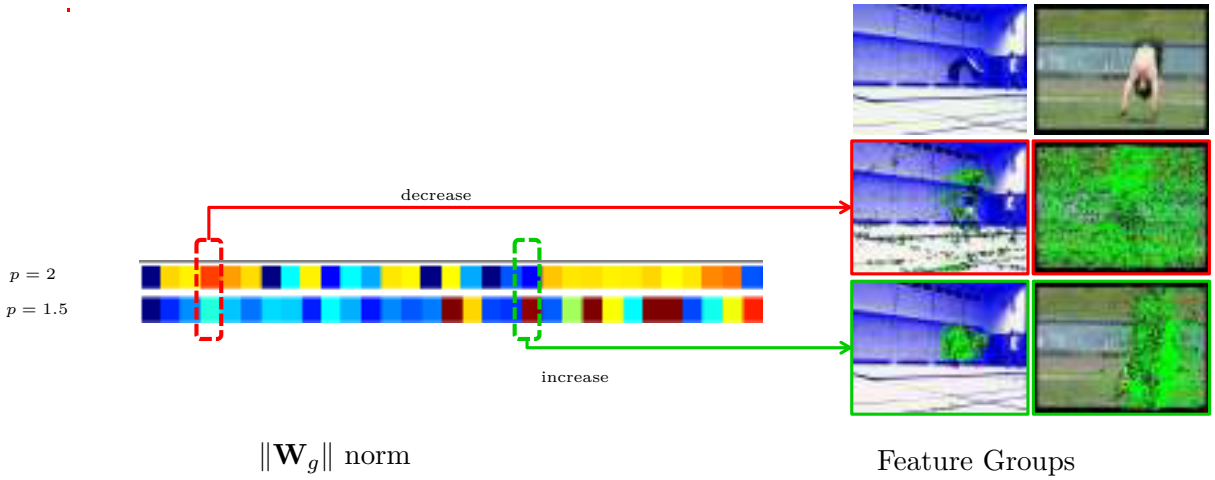


Figure 6-14: Evaluation sparse feature weighting regularizer for the "Flic Flac" action on HMDB. On the left, $\|\mathbf{W}_g\|_2$ are displayed, for $p = 2$ or 1.5. On the right, features corresponding to two $\mathbf{W}_g$ groups are shown.

to 51.8, 2.1% relatively. WSVM benefits from $\mathbf{W}$ structure to learn task-specific saliency layout. For $p \leq 1$, we observe a performance decrease. In this case, $\mathbf{W}$ becomes too sparse, selecting too few *structural primitives*. It justifies the use of a $\|.\|_{2,p}$ regularizer, allowing the control of sparsity, instead of a more rigid $\|.\|_{2,1}$ norm.

Figure 6-14 illustrates the impact of the sparsity parameter p for the HMDB "Flic Flac" action showing that p allows discriminative features to increase in importance while reducing the impact of noisy feature groups.

6.6 Conclusion

This chapter has introduced a new space-time invariant pooling scheme leading to a video attention context that leverages the video space-time information while remaining invariant to global space-time transformations. The attention context, or structural-BoWs, identifies prominent regions in videos content through *motion*, *illumination* and *cornerness* saliencies, leading to a "video-based" segmentation scheme. We also benefit from the WSVM to automatically learn, in a bottom-up fashion, the optimal saliency layout associated with an action. We showed through an extensive experimentation that:

- the distribution of discriminative information is non-uniform in the saliency domains. Taking into account different saliencies through the content-based pooling increases the performance by 16.5% on average, comparatively to the BoW representation on the HMDB dataset.

- being robust to the space-time variance is of prime importance for action recognition. Our video attention context combining the motion, light, cornerness saliencies constantly outperform the fixed spatial segmentation on the KTH, UCF50 and HMDB. The performance gain reaches 10% on HMDB.

- our content based pooling and spatial pooling are complementary. Their combination reaches a further gain of 7% on HMDB.

- WSVM allows the selection of the most discriminative structural-primitives associated with an action. Using the sparsity regularizer allows a performance gain of 2% compared to a non-sparse SVM classifier.

Chapter 7

Evaluation of Multiple Contexts Representation

This chapter proposes an evaluation which balances multiple contexts for action annotation. We take advantage of the classification framework introduced in Chapter 3. We consider the different contexts developed in this thesis as well as some pre-existing ones. We validate our approach on the UCF101 [178] and HMDB [95], which are currently two of the most challenging datasets for action recognition.

Most noticeably on the HMDB dataset, our system shows a performance improvement of 28% and 21% compared to the traditional bag-of-words and spatial bag-of-words.

7.1 Experimental Setting

We consider 9 contexts: STIP-BoW, Traj-Cov, Traj-BoW, Traj-spBoW, Traj-agBoW, Traj-Cornerness, Traj-Light, Traj-Motion and SEM. The different contexts are summarized in Table 7.1.

STIP-BoW, Traj-BoW and Traj-Cov are *Feature Contexts*. STIP-BoW relies on space-time interest points [101] while Traj-BoW and Traj-Cov take advantage of dense trajectories [197]. A BoW model [170] is used by STIP-BoW and Traj-BoW to ag-

Context		Local Feature	Aggregation
Feature	STIP-BoW	STIP [100]	BoW [170]
	Traj-BoW	Trajectory [197]	BoW [170]
	Traj-Cov	Trajectory [197]	Covariance (Chap. 4)
Space-Time	Traj-spBoW	Trajectory [197]	Fix Grids [102]
	Traj-agBoW	Trajectory [197]	Adaptive Grids (Chap. 5)
	Traj-(Co+Li+Mo)	Trajectory [197]	Content-based Pooling (Chap. 6)
Semantic	SEM	Learned	DCNN [98]

Table 7.1: Context Synopsis.

gregate the local features. As in the previous chapters, BoW aggregation is designed with LLC-coding (and a visual vocabulary of size 4000) and max-pooling [196]. Traj-Cov benefits from the average covariance pooling (see Chapter 4). It characterizes the linear dependencies of local trajectory descriptors.

Traj-spBoW, Traj-agBoW , Traj-Cornerness, Traj-Light, Traj-Motion are *Space-Time Contexts*. In addition to the video content, they leverage the local feature space-time localizations. Traj-spBoW performs fix grid spatial pooling [102]. It divides the video volume using a 2x2x2 fix segmentation grid and compute one BoW per grid cell. Traj-agBoW learns 16 segmentation grids directly from the video data (see Chapter 5). Traj-Cornerness, Traj-Light, Traj-Motion are attention contexts using respectively Cornerness, Light and Motion saliencies (see Chapter 6). In the following we actually consider the different attention contexts jointly, we denote their combination as Traj-(Co+Li+Mo).

SEM is a *Semantic Context* which was first introduced by Lan *et al.* [98]. Authors of [98] learn classifiers capturing visual appearance of 1000 objects based on the ImageNet dataset [35] using Deep Convolutional Neural Networks (DCNN). In this context, each video is then represented by a vector of size 1000 characterizing the presence or absence of each object. We use the same SEM settings than Lan [98].

7.2 Individual Context Evaluation

In a first experiment, we evaluate the different contexts individually. All contexts except Traj-Cov are associated with a linear SVM model. Traj-Cov relies on a bi-linear SVM with 15 compound components (see Chapter 4). The regularization parameter λ of both SVMs is set to 0.1 (see Chapter 3).

Traditional evaluation settings established by Kuehne [95] are used for HMDB. For UCF101, we rely on the train and test splits provided by the international THUMOS evaluation [188].

Context		Accuracy
Feature	STIP-BoW	27.6
	Traj-BoW	41.6
	Traj-Cov	48.3
Space-Time	Traj-spBoW	44.3
	Traj-agBoW	46.8
	Traj-(Co+Li+Mo)	**48.5**
Semantic	SEM	25.0

Table 7.2: Average Accuracy of the different contexts on the HMDB dataset.

Table 7.2 reports the context average accuracies on HMDB. We observe that the different contexts developed in this thesis (Traj-Cov, Traj-AG and Traj-(Co+Li+Mo) obtain competitive results. They achieve the 3 best individual performances on HMDB.

Traj-agBoW and Traj-(Co+Li+Mo) capture the same type of information. They both model the local feature space-time distribution using segmentation schemes which take into account the underlying video data. While the Traj-AgBoW segmentation (relying on Adaptive Grid) is action-specific, Traj-(Co+Li+mo) takes advantage of content-based pooling to provide video-specific segmentation. We notice that a video-specific segmentation scheme leads to better results. On HMDB, Traj-(Co+Li+Mo) outperforms by 4% the Adaptive Grids (48.5 compared to 46.8). Consequently, to limit the dimensionality of our video representation, we choose to use the

video-specific content-based pooling instead of action-specific Adaptive Grids when combining several contexts.

Regarding the other contexts, Table 7.2 confirms the importance of long-term temporal information. Traj-BoW using trajectory features (which captures long-term motion region) outperforms by 50% the short-time duration STIP features.

The SEM representation obtains an average accuracy of 25.0 which is the lowest score relatively to the other contexts. Despite having a semantic interpretability, SEM context characterizes only appearance information. It learns concept appearances directly from static images. It therefore discards the video motion. HMDB has been specifically designed to provide a dataset whose human action categories mainly differ in motion rather than human static pose [95]. Actions therefore have strong appearance variability. It limits the performances of contexts such as the SEM context which doesn't take into account the motion information.

Context		Accuracy
Feature	STIP-BoW	73.1
	Traj-BoW	85.4
	Traj-Cov	**90.0**
Space-Time	Traj-spBoW	87.8
	Traj-agBoW	88.2
	Traj-(Co+Li+Mo)	89.8
Semantic	SEM	79.3

Table 7.3: Average Accuracy of the different contexts on the UCF101 dataset.

Table 7.3 reports each individual context result on the UCF101 dataset. Traj-Cov achieves the best performances with 90.0 outperforming by 5.3% the Traj-BoW representation. We can notice that the SEM context obtain better performance on UCF101 than HMDB. It obtains a gain of 8.4% compared to the STIP-BoW representation. It shows that appearance information is more discriminative on UCF101 than on HMDB.

7.3 Context Combination Evaluation

In a second experimentation, we investigate the combination of contexts for action recognition.

7.3.1 Combination Model

Contexts are combined through the WSVM classification model introduced in Chapter 3. We briefly remind the WSVM objective function.

Let $\mathbf{D} = \{\mathbf{X}, \mathbf{Y}\}$ be a training dataset composed by N videos. $\mathbf{X} = \{\mathbf{X}_1, ..., \mathbf{X}_N\}$ are the video intermediate representation and $\mathbf{Y} \in \{0,1\}^N$ are the binary labels. Each video intermediate representation $\mathbf{X}_i$ is the concatenation of C contexts, *i.e.* $\mathbf{X}_i = [\mathbf{X}_i^1, \ldots, \mathbf{X}_i^C]$. WSVM model minimizes the following objective function

$$O(\mathbf{W}, \mathbf{D}) = \sum_{i=1}^{N} L(Y_i, \hat{Y}_i \sum_{c=1}^{C} M_c(\mathbf{W_c}, \mathbf{X}_i^c)) + \lambda ||\mathbf{W}||_{2,p}. \tag{7.1}$$

In (7.1), $\hat{Y}_i$ is the predicted label, M_c are the model associated with each context and $\mathbf{W}_c$ the model parameters. All contexts, even Traj-Cov, use linear model for this experiment (as specified in the next section).

Two parameters control the WSVM model: the regularization weight λ and the sparsity parameter p. Following Chapter 3, the WSVM regularization parameter λ is set to 0.1. When it is not explicitly specified, the sparsity parameter p is set to 1.5. We study the impact of the sparsity parameter p in 7.3.4.

7.3.2 Technical Details

Table 7.4 reports dimensions and memory footprints associated with each context. If we consider all the contexts at the same time, we obtain a video representation of size 302476. An augmentation of the representation dimension leads to an increase of the learning algorithm memory consumption. Indeed, a WSVM model is used for classification. WSVM relies on a batch learning process, *i.e.* it needs to load all the

Context	Dimension	Memory Footprint	
		HMDB 6676 Videos	UCF101 13320 Videos
SEM	1000	51M	101M
STIP-BoW Traj-BoW	4000	206M	406M
Traj-Cornerness Traj-Light Traj-Motion	28000	1446M	2845M
Traj-spBoW	32000	1652M	3252M
Traj-Cov	181476	9367M	18442M

Table 7.4: Context memory footprint.

training examples to perform the learning.

Loading all the video contexts at once in memory requires more than 60G for UCF101. To limit the memory footprint, we choose to apply a Principal Component Analysis (PCA) to reduce the dimensionality of the Traj-Cov context. Using PCA, we reduce the dimension of the Traj-Cov to 10000. After PCA, Traj-Cov vectors are fed to a linear model. Another possible strategy to limit the memory consumption which has not been investigated in this thesis, would be the consideration of stochastic-based learning algorithm. Stochastic learning only requires the loading of one training sample in memory at a time. It has been demonstrated that stochastic learning achieves the same performances than batch learning while strongly reducing the computational and memory costs [2].

7.3.3 Combination Results

Table 7.5 reports the performances of video representations that leverage multipe contexts. We evaluate two different combinations of contexts:

- Traj-Combination which considers multiple contexts using dense trajectory local features. Traj-Combination is composed by Traj-BoW, Traj-spBoW, Traj-Cov and Traj-(Co+Li+Mo) contexts.

Context Combination	Accuracy
Traj-Combination	**53.3**
All-Combination	52.8
Best individual context: (Traj-(Co+Li+Mo))	48.5

(a) Average Accuracy on the HMDB dataset.

Context Combination	Accuracy
Traj-Combination	91.7
All-Combination	**93.1**
Best individual context: Traj-Cov	90.0

(b) Average Accuracy on the UCF101 dataset.

Table 7.5: Combination results.

- All-Combination which considers multiple contexts using multiple features. All-Combination considers all the contexts except Traj-agBoW (see 7.2). It therefore leverages the trajectory, STIP and DCNN features.

Table 7.5 demonstrates that multiple contexts-based representation are indeed useful for concept recognition. On HMDB, Traj-Combination achieves a gain of 9.5% relatively to the best individual context (Traj-(Co+Li+Mo)). The combination of trajectory-based context with STIP-BoW and SEM does not help to further improve the performances. We report the Spearman's ρ factor (4.23) between STIP-BoW, SEM and All-Trajectory contexts. It shows that STIP-BoW and SEM are strongly correlated with the All-Trajectory, they don't add discriminative information. Indeed, HMDB has been designed to provide a dataset whose action categories differ in motion rather than in appearance [95]. However both STIP-BoW and SEM mainly have a limited modeling of motion information.

On UCF101, Traj-Combination obtains a gain of 1.8% compared to the best individual context. On this dataset, adding STIP-BoW and SEM contexts does lead to another improvement. All-Combination reaches a gain of 3.2% compared to the best individual context. Appearance is more discriminative on UCF101 than HMDB. Hence, STIP-BoW and SEM contexts tends to capture complementary information to the Traj-Combination as Table 7.6 shows.

	STIP-BoW	SEM
Traj-Combination	92.4	95.1

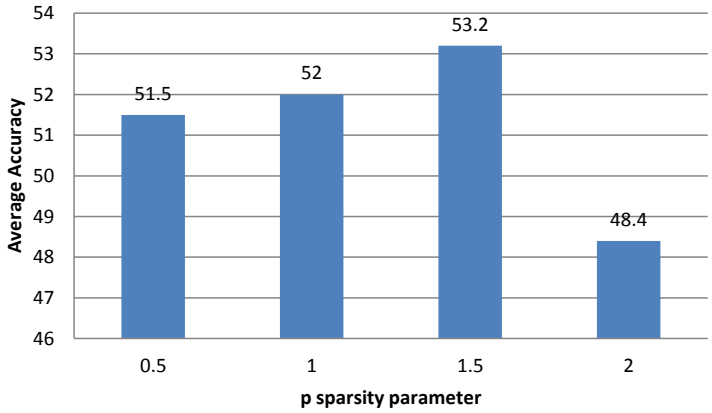
(a) HMDB dataset.

	STIP-BoW	SEM
Traj-Combination	86.4	74.1

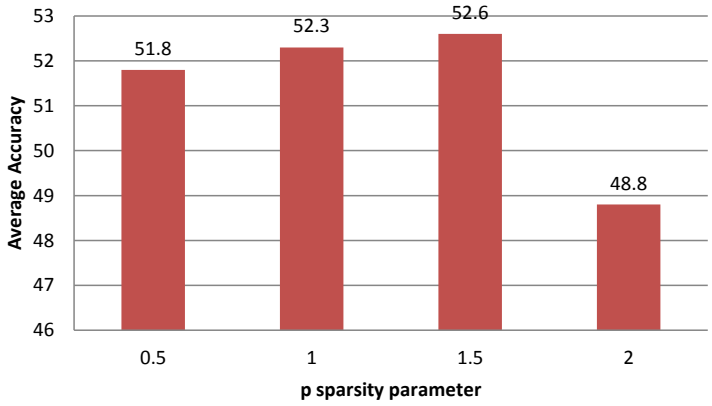
(b) UCF101 dataset.

Table 7.6: Spearman's ρ factor.

7.3.4 Sparsity Impact



(a) Traj-Combination



(b) All-Combination

Figure 7-1: Evaluation of the sparsity parameter p on the HMDB dataset.

We finally evaluate the impact of the sparsity parameter p on the classification performances. Figure 4-11 reports the average accuracy given different value of p for

the WSVM model on the HMDB dataset. It shows that inducing sparsity $p < 2$ in our action model does improve the performance for both Traj-Combination and All-Combination. For All-Combination, setting $p = 1.5$ obtains a gain of 7% compared to a traditional non-sparse SVM, *i.e.* $p = 2$. Selecting only the more relevant contexts accordingly to an action is therefore primordial to achieve good performance at classification. Indeed, adding sparsity in our classification model allows emphasizing the context that fit at best our action. As for the attention context combination (see Section 6.5.4), for $p \leq 1$, we observe a performance decrease. In this case our model becomes too sparse discarding discriminative information.

7.4 Comparison with State-of-art

HMDB		UCF101 25 fold	
Kuehne *et al.* [95]	23.0		
Sadanand *et al.* [161]	26.9		
Cao *et al.* [26]	27.8		
Jiang *et al.* [81]	40.7		
Wang *et al.* [197]	46.6		
Shi *et al.* [167]	47.6		
Jain *et al.* [70]	52.1		
Wang *et al.* [198]	**57.2**	Soomro *et al.* [178]	44.5
Our approach	53.3	Our approach	**87.7**

Table 7.7: Comparison with state-of-the-arts. Average Accuracy is reported.

Table 7.7 compares our approach with the state-of-the-art on the UCF101 and HMDB dataset. At the time of this dissertation redaction, no results have yet been published using the THUMOS evaluation setting. We therefore adopt a 25 group-wise cross-validation to compare with previous works [178].

On HMDB, we achieve the second best performance with an average accuracy of 53.3. Our approach underperforms [198]. Authors of [198] use dense trajectories along with camera motion estimation to add motion-compensated trajectories in the

video signature. There approach is complementary to our work and their combination could lead to further performance improvement.

Nevertheless our approach achieve a gain of 14.6% compared to fix grid spatial pooling using traditional dense trajectories without motion compensation. It therefore shows the relevance of multiple-contextual approaches for action recognition.

To our knowledge, Soomro *et al.* [178] have been the only one reporting their performances on UCF101 using a 25 group-wise crossvalidation. Our approach obtains a strong gain of 92% comparatively to their works. Authors of [178] use only STIP BoW representation to perform action recognition. By contrast, we leverage multiple-features and multiple contexts.

7.5 Conclusion

In this chapter, we proposed an evaluation of a multiple-context system for action annotations in videos. Our system leverages 9 different contexts based on STIP, trajectory features or deep neural network [98]. We validate our approach on the UCF101 [178] and HMDB [95], which are currently two of the most challenging datasets for action recognition.

We draw the following conclusion from our experimentations:

- Combining several contexts is useful for action recognition. Leveraging several context reach allows to a performance gain up to 9.5% on HMDB and 3.2% on UCF101 comparatively to the best individual context.

- Sparsity helps to improve the performance when combining several contexts. Adding sparsity in our WSVM model obtains a performance gain of 7% relatively to a non-sparse classification model.

Chapter 8

Conclusion

To conclude our work, we summarize our main contributions and discuss interesting directions for further research in this field.

8.1 Key Contributions and Immediate Perspectives

The definition of video intermediate representation is primordial for automated action recognition (see Chapter 3). Such representation needs to highlight discriminative information associated with the action concepts while discarding irrelevant detail, in order to determine what falls in which category. Throughout this dissertation, we have investigated 3 new video representations. More specifically, we proposed 3 pooling operators (Covariance, Task-Specific Space-Time, Content-based Pooling), leading to new video visual contexts. Our experimental study demonstrated that:

- higher-order local features statistics refine the video representation discriminative power;
- local feature space-time information is action dependent;
- preserving the space-time invariance while leveraging the local-features space-time localization improves the concept annotation performances.

We also proposed a novel classification framework identifying the relevant contexts given an action. Using this framework, we showed that multiple contexts representation improve the concept annotation performances. Those different contributions

	State-of-art		Thesis		Gain
UCF-101 [178]	44.5	[178] 2013	mutiple contexts	**87.2**	92%
HMDB [95]	57.1	[198] 2013	multiple contexts	**53.3**	-
UCF-50 [157]	84.5	[198] 2013	attention contexts	**92.7**	9%
UCF-Youtube [113]	84.0	[197] 2011	space-time context	**86.3**	4%
KTH [162]	94.5	[49] 2011	covariance context	**95.5**	1%
UT-Interaction 1 [160]	84.0	[144] 2012	space-time context	**91.3**	9%
UT-Interaction 2 [160]	86.0	[144] 2012	space-time context	**95.0**	11%

Table 8.1: Overview of the main thesis results. Average Accuracy is reported.

achieved to competitive results as Table 8.1 reports. We detail each contribution in the remaining of this section.

8.1.1 Covariance Context

Contribution (Chapter 4) Local descriptors capture different aspects of the visual content (appearance, motion, acceleration, *etc*). Existing video representations generally don't explicitly consider their linear dependency. However, such descriptor linear dependencies can bring discriminative information. Descriptor covariance, for instance, captures mid-level patterns that characterize jointly the motion and appearance in video. Covariance is especially relevant in our case since actions are jointly defined by movements and appearances. We therefore proposed a novel context which captures the descriptor covariance information.

Our finding shows that covariance of local descriptors enables further discriminative capability. On the HMDB dataset, covariance context outperforms the BoW representation by 16%. In addition, when considering the covariance and BoW representation a performance gain of 22% is obtained with respect to the sole BoW. Covariance and BoW representations are therefore complementary.

Relying on higher-order statistics, covariance allows designing a video representation with a strong discriminative power, but this context tend to be sensible to outlier features present in a video.

Immediate Perspectives Pooling operator computes statistics from the local features to characterize a visual content. While bag-of-words [170] and other visual vocabulary based approaches (Fisher Kernel [145], VLAD [73]) has demonstrated to be competitive pooling operators, we saw in this thesis that covariance moment should also be taken into account. We limited our study to the covariance, but, many other statistical moments can be considered such as skewness, kurtosis or quantile statistics. It is yet unclear how to choose a pooling operator given the local feature distribution and the visual task at end.

8.1.2 Task-Specific Space-Time Context

Contribution (Chapter 5) Local features space-time localization conveys discriminative information for action recognition [106]. However, most of local representation approaches have a limited modeling of the video space-time context. State-of-art solutions rely on statically defined segmentation grids to embed space-time information in a bag-of-words model. They use the same segmentation grids for all the actions. Due to their static aspect, there is no guarantee that the segmentation grids will fit the space-time distribution associated with an action. To tackle this issue, we introduced an action-specific space-time context through the adaptive grids. Our approach learns the action space-time shape directly from the training dataset, adaptive grids are able to coarsely follow the action through time in videos.

We evaluate our proposal on 4 standard datasets. On average, our adaptive grids obtain a performance gain of 9.5% compared to traditional approaches which use predefined and fix segmentation layouts.

Our approach is especially useful to characterize the space-time context of actions with strong localization variability. In such case, adaptive grids follows coarsely the action main localization through time in the videos. It does not lead to improvement, comparatively to fix grid, for actions with stable localization over time in video.

Immediate Perspectives Task-Specific Space-Time Context allows a better modeling of the space-time information between the local features. However, our current

approach discards the space-time regions co-occurence information. We do not encode the relative arrangement of the different space-time regions. Space-time regions co-occurence information could potentially bring additional discriminative power to the representation. Indeed, an action is defined as a combination of local space-time regions with specific appearances and motions. Space-time regions co-occurences would characterize how the different local region localizations jointly evolve in videos.

8.1.3 Attention Context

Contribution (Chapter 6) Modeling the space-time information in video generally implies a loss of the space-time invariance. Retaining space-time invariance is critical for action representation as actions know dramatic space-time variances. We proposed a novel representation that leverages the video space-time information while remaining invariant to the global space-time transformations. This representation takes advantage of saliency functions to identify prominent regions while inheriting from their invariance properties. In particular, we investigated *motion*, *illumination* and *cornerness* saliencies

We showed through an extensive experimentation that the distribution of discriminative information is non-uniform in the saliency domains. Taking into account the saliency information increases the performance by 16.5% on average, comparatively to the BoW representation. We also showed that our attention outperforms traditional space-time approaches, up to 24%.

Being robust to the space-time variance is therefore of prime importance for action recognition. However, this performance gain comes with a computational cost since the attention map needs to be computed for each video frame.

Immediate Perspective In this thesis, we estimated the attention map using 3 different bottom-up saliency functions which extract structural properties of the video. In our case, we focused on *motion*, *illumination* and *cornerness* properties of the visual content. Other saliencies exist in the literature [21]. Our representation could therefore be enriched with new saliencies capturing complementary structural

properties. Color-based saliencies or saliencies taking into account the temporal dimension of the videos would likely bring some complementary information to the representation.

Our approach also learns which bottom-up saliency functions are discriminative given an action. However, there is no guarantee that a predefined bottom-up saliency highlights discriminative regions of a particular action. Differently, we could learn the saliency function directly from the training dataset [166]. We could infer a saliency functions maximizing the classification performances, using appearance information to infer which regions are discriminative given an action.

8.1.4 Multiple-Contexts Classification Framework

Contribution (Chapter 3) All the contexts are not equivalent. Some contexts are more informative about the presence of a concept in multimedia content than others. Using this insight, we propose a learning framework that automatically determines which are the relevant contexts associated with a concept. Our model weights automatically the relevant contextual information associated with a concept. We leverage group-sparse regularization to limit the number of contexts used to model a concept. By focusing only on a few contexts, we take advantage of intermediate representations which describe at best the concept of interest while discarding irrelevant and noisy representation.

Using several contexts, our multiple contextual annotation frameworks leads to a gain of 28% compared to the sole BoW representation (see Chapter 7).

Immediate Perspective In this thesis, we showed that choosing concept-specific representations improves the action annotation performances. However, the optimal representations may change even within a concept class. Indeed, intra-class videos are subject to variation. For instance, they can be recorded under different camera viewpoints. Intra-class variation can possibly impact which representation fits at best the video. To overcome this issue, we could infer which representations fit at best the current video given the concept we want to recognize. Representations

weighting would use both concept and video information leading to a model that selects video-specific representations. Exemplar-SVM learns one representation per video, maximizing its distinctiveness. One could add sparsity constraint to Exemplar-SVM model [120] applied on several contexts in order to select the most distinctive. However, such approach would be computationally intensive.

8.2 Future Directions

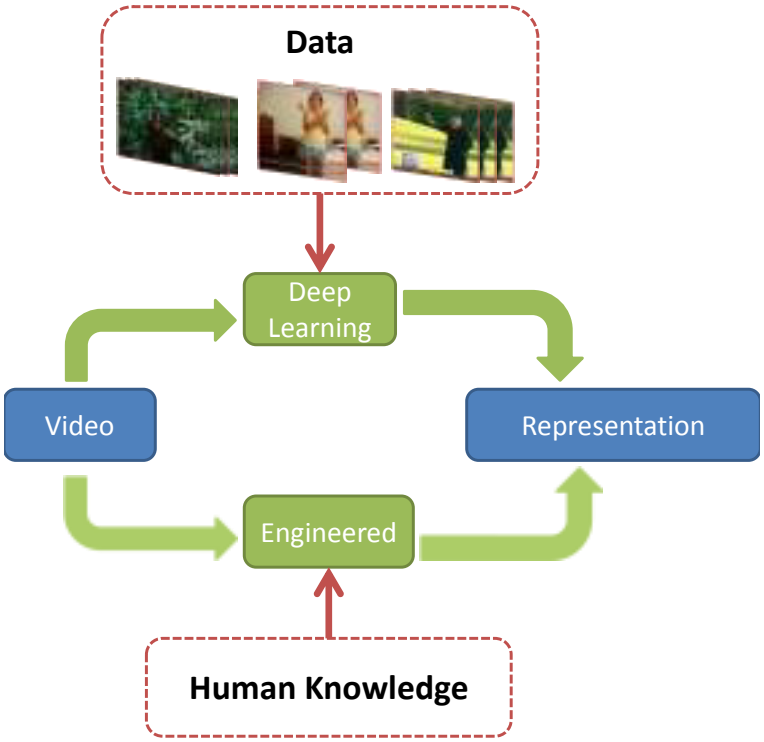


Figure 8-1: Deep Learning vs Engineered Representations.

Finding an optimal video representation is still a challenging problem. In this dissertation, we saw that adding represention generally increases the classification performances. Different actions have different optimal representations. Hence, by considering multiple representations, we take advantage of their complementarity and improve the automated concept annotation performances. However, the design of representation requires a lot of engineering effort to achieve competitive performances. It is a costful and time consuming operation. Moreover, there is no guarantee that

an engineered representations will provide a good description for a particular action. Deep Learning could be an interesting direction to solve those issues. Contrary to engineered representation Deep Learning learn directly the representation from the training video data (see Figure 8-1). It reduces the representation design costs since a specific representation is automatically learned for each concept class.

Despite having obtained some encouraging result in image recognition [40, 94], their extension to videos remains an open issue. It raises the question on how to integrate temporal information in the Deep Learning model. Le *et al.* [107] have proposed a Deep Learning model for action recognition. However, they only consider short term motion information. Their approach does not outperform engineered trajectory feature [197], describing long-term motion information.

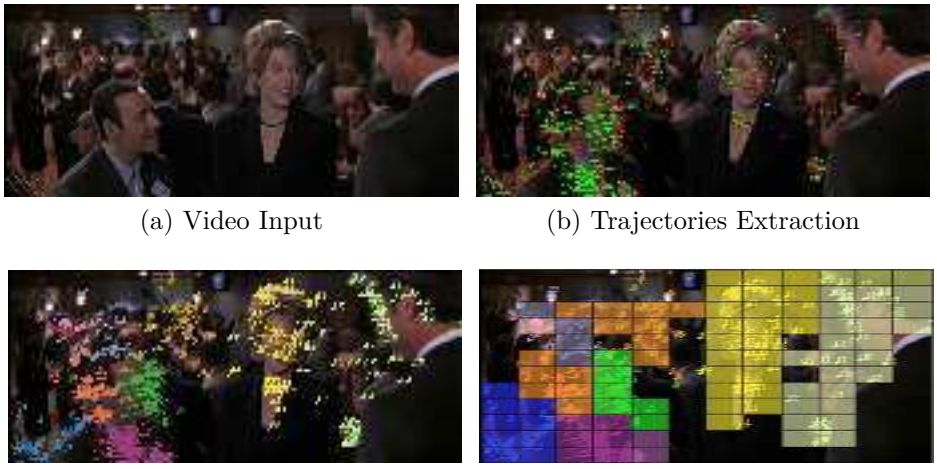
Deep Learning also requires large training dataset to learn relevant representations. Video datasets are steadily growing in size. During the last decade, they have evolved from few hundred videos [162] to several thousand [95, 157]. Despite their scale increase, datasets still remain limited either in term of concept categories or by their video numbers. Recent datasets [171] consider up to 362 different categories. However, it has been shown that a concept annotation system needs at least 5000 visual concepts to achieve retrieval accuracy comparable to text search engine [59]. In addition, action recongition datasets have typically around one hundred videos per action category. Due to the high-variability of the action visual appearance, one hundred videos tends to be too limited to fully characterize an action.

Although we identified some possible bottlenecks associated with Deep Learning approaches, their impressive results in context of static image annotaiton [94] make them worth considering as a future research direction.

Appendix A

Video Segmentation

In this appendix, we present the video segmentation algorithm used in Chapter 5 to extract space-time region from videos. Our segmentation perform a trajectory clustering of trajectory features in order to identify space-time region in videos.



(a) Video Input (b) Trajectories Extraction

(c) Trajecotries Clustering (d) Video Segmentation

Figure A-1: Illustration of our video signatures computation. First local trajectory features are extracted from a video input (figure A-1b). Then trajectory cluster are computed through clustering (figure A-1c). Finally, we take advantage of the tunnel features spatio-temporal positions to obtain our final video segmentation (figure A-1d).

Given a video $\mathbf{V}$, we want to obtain a set of spatio-temporal regions. We identify those regions in $\mathbf{V}$ relying on trajectory features. Each segmentation region is trajectory cluster representing connected areas of the spatio-temporal video volume having

consistent motion.

A.1 Gibbs Point Process Model For Segmentation

We consider a set of dense trajectory features $\mathbf{T} = \{\mathbf{t}_i\}_{i\in[1,M]}$ [196] extracted from a video $\mathbf{V}$. Our goal is to compute compute a set of trajectory clusters $\mathbf{O} = \{\mathbf{o}_j\}_{j\in[1,Q]}$, where each $\mathbf{o}_j$ is a subset of $\mathbf{T}$ that respects some motion and spatial locality constraints. Considering $\mathbb{O}$, the set of possible cluster configuration, our goal is to find $\mathbf{O} \in \mathbb{O}$ that maximizes the joint probability law P:

$$\hat{\mathbf{O}} = \arg\max_{\mathbf{O}\in\mathbb{O}} P(\mathbf{O}, \mathbf{T}) = \arg\max_{\mathbf{O}\in\mathbb{O}} P(\mathbf{O})P(\mathbf{O} \mid \mathbf{T}). \tag{A.1}$$

Modeling the prior $P(\mathbf{O})$ is rather a complicated task, and often need some restrictive assumptions. Instead of modeling our segmentation as (A.2), we opt for discriminate model by neglecting the prior $P(O)$ to obtain a classic MAP (Maximum A Posterior) estimation problem:

$$\hat{\mathbf{O}} = \arg\max_{\mathbf{O}\in\mathbb{O}} P(\mathbf{O} \mid \mathbf{T}). \tag{A.2}$$

To cope with a variable number of trajectory clusters in video, the probability law P is modeled as a Gibbs point process [158] . Gibbs point process model is a natural extension of the Markov Random Field (MRF) [16, 45]. It allows the modeling, within a stochastic framework, of a random number of objects, avoiding the limitations introduced by the static aspect of MRF graph [36]. A Gibbs point process defined a density which models the likelihood of a realization $\mathbf{O}$:

$$P(\mathbf{O} \mid \mathbf{T}) = \frac{h(\mathbf{O})}{\int_{\mathbb{O}} h(o)do}. \tag{A.3}$$

In (A.3) h is the Gibbs density while $\int_{\mathbb{O}} h(o)do$ is the normalization constant. By maximizing $h(\mathbf{O})$, we are maximizing $P(\mathbf{O} \mid \mathbf{T})$. It appears clearly that the expression of the density $h(\mathbf{O})$ is a key aspect in our approach.

A Gibbs density is defined with a potential U which represents the cost associated to a cluster configureation:

$$h(\mathbf{O}) = e^{-U(\mathbf{O})}. \tag{A.4}$$

We want to obtain clusters that respect spatial locality and motion coherence constraints. To express those constraints, we model U using a combination of attraction (a) and repulsion (r) potentials [110]:

$$a(\mathbf{o}_j) = \sum_{\mathbf{t}_i \in \mathbf{o}_j} \sum_{\mathbf{t}_k \in \mathbf{T} \setminus \{\mathbf{o}_j\}} e^{-\lambda d(\mathbf{t}_i, \mathbf{t}_k)}, \qquad r(\mathbf{o}_j) = \sum_{\mathbf{t}_i \in \mathbf{o}_j} \sum_{\mathbf{t}_k \in \mathbf{o}_j} 1 - e^{-\lambda d(\mathbf{t}_i, \mathbf{t}_k)}. \tag{A.5}$$

In (A.5), $\mathbf{o}_i$ is a cluster object, λ is a constant set to 0.1, and d a function computing the divergence between two trajectory. Our function d is inspired from [23]:

$$d(\mathbf{t}_i, \mathbf{t}_j) = \frac{1}{|l - f|} \sum_{k=f}^{l} |\mathbf{p}_k^{\mathbf{t}_i} - \mathbf{p}_k^{\mathbf{t}_j}||\mathbf{m}_k^{\mathbf{t}_i} - \mathbf{m}_k^{\mathbf{t}_j}|, \tag{A.6}$$

where f (respectively l) is the first (respectively last) common frame between $\mathbf{t}_i$ and $\mathbf{t}_j$, $\mathbf{p}_k^{\mathbf{t}_i}$ (respectively $\mathbf{m}_k^{\mathbf{t}_i}$) the $\mathbf{t}_i$ trajectory position (repectively motion vector) at the frame k. d ensures in (A.5) that only trajectory spatially close and having similar motion will yield high similarity values. Our final potential term becomes

$$U(X) = \sum_{\mathbf{o} \in X} \alpha_1 a(\mathbf{o}) + \alpha_2 r(\mathbf{o}). \tag{A.7}$$

Here, α_1, α_2 are potential fusion coefficients that are empirically determined. In practice, we set $\alpha_1 = 1$ and $\alpha_2 = 0.1$

A.2 Optimization

We need to maximize $P(\mathbf{O} \mid \mathbf{T})$ to find the optimal cluster configuration accordingly to $\mathbf{T}$. $P(X \mid F) = \frac{h(X)}{\int_{N^X} h(x)dx}$. Due to the intractable normalizing constant, it is not possible to maximize $P(X \mid F)$ directly. We take advantage of the Metropolis-

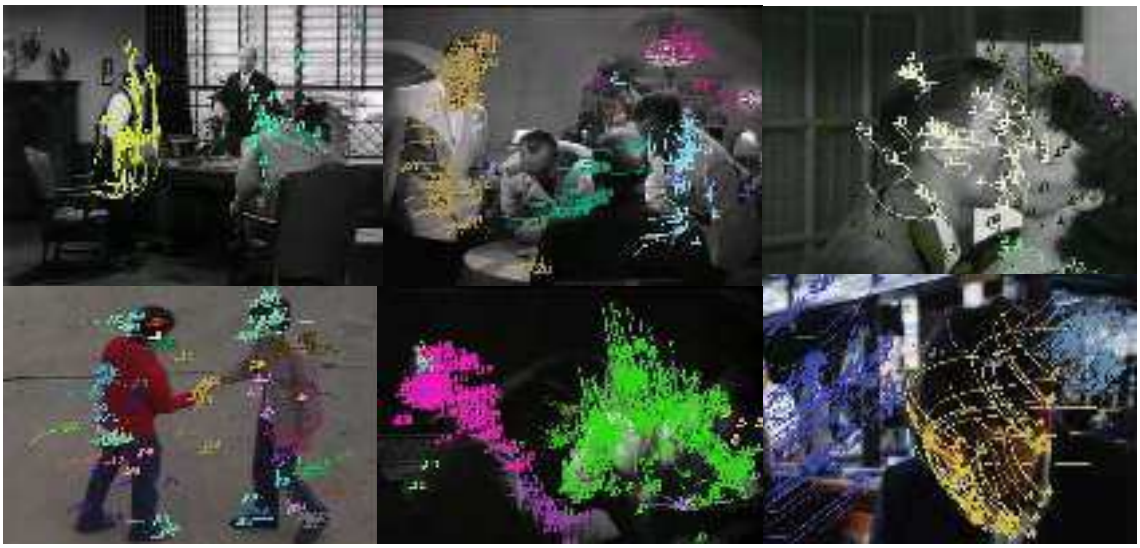


Figure A-2: Some trajectory segmentation results.

Hasting-Green that allows simulating a point process model specified by unnormalized density through the use of proposal distribution kernels.

Metropolis-Hasting-Green (MHG) algorithm is a MCMC technique[47] that relies on a birth and death sampler to handle the variable dimensions of the different point process configurations. MHG uses several sampler kernels $Q_i(X, Y), i \geq 0$ that update the state of our point process configurations. The kernel captures the state-transition distribution regarding a certain update operation. At each step, a sampling kernel Q_i is selected with a probability p_i. We update our point process X to Y using the kernel sampling distribution $Q_i(X, .)$. The point process state modification is then accepted with probability $max(1, R)$ where $R = \frac{h(Y)Q(X,Y)}{h(X)Q(Y,X)}$ is the green ratio indicating the "likelihood" of the sampling.

Three different sampling kernels Q_1, Q_2 and Q_3 are used by our trajectory grouping-algorithm. Q_1 is the birth/death kernel that creates or removes a cluster at a random position. Q_2 is the add/del kernel that adds or deletes a trajectory in a cluster. Finally, Q_3 is the fuse/divide kernel that fuses two close clusters or divides one inhomogeneous cluster. To improve the convergence speed of the algorithm, those kernels are driven by the spatial distance between trajectories. For instance, for the addition of a new trajectory in a cluster we will only consider the n-closest trajectories to the cluster elements, and, the fusion operation will be only considered for clusters which

elements are spatially close.

Since, video segmentation was not a primary objectif of this dissertation, we did not quantitavelly evaluate our segmentation algorithm. Figure A-2 provides result examples of our segmentation algorithm.

Bibliography

[1] J. Aggarwal and M. S. Ryoo. Human activity analysis: A review. *ACM Computing Surveys (CSUR)*, 43(3):16, 2011. 43

[2] Z. Akata, , F. Perronnin, Z. Harchaoui, and C. Schmid. Good Practice in Large-Scale Learning for Image Classification. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2013. 59, 60, 82, 105, 190

[3] P. Atrey, M. Hossain, A. El Saddik, and M. Kankanhalli. Multimodal fusion for multimedia analysis: a survey. *Multimedia Systems*, pages 1–35, 2010. ISSN 0942-4962. 44, 62

[4] F. Bach, R. Jenatton, J. Mairal, and G. Obozinski. Optimization with sparsity-inducing penalties. *arXiv preprint arXiv:1108.0775*, 2011. 72

[5] F. Bach, R. Jenatton, J. Mairal, and G. Obozinski. Structured sparsity through convex optimization. *Statistical Science*, 27(4):450–468, 2012. 72

[6] F. R. Bach, G. R. Lanckriet, and M. I. Jordan. Multiple kernel learning, conic duality, and the smo algorithm. In *Proceedings of the twenty-first international conference on Machine learning*, page 6. ACM, 2004. 59

[7] L. Ballan, M. Bertini, A. Del Bimbo, and G. Serra. Video event classification using bag of words and string kernels. *Image Analysis and Processing–ICIAP 2009*, pages 170–178, 2009. 57

[8] L. Ballan, M. Bertini, A. Del Bimbo, L. Seidenari, and G. Serra. Event Detec-

tion and Recognition for Semantic Annotation of Video. *Multimedia Tools and Applications*, pages 1–24, 2010. ISSN 1380-7501. 43

[9] N. Ballas, B. Delezoide, and F. Prêteux. Trajectories based descriptor for dynamic events annotation. In *ACM workshop on Modeling and representing events*, pages 13–18. ACM, 2011. 51, 75, 95

[10] N. Ballas, B. Delezoide, and F. Prêteux. A new point process model for trajectory-based events annotation. In *Proceedings of SPIE*, volume 8300, page 83000J, 2012. 137, 138

[11] A. L. Bao, S.-I. Yu, Z.-z. Lan, A. Overwijk, Q. Jin, B. Langner, M. Garbus, S. Burger, F. Metze, and A. Hauptmann. Informedia@ trecvid 2011 multimedia event detection, semantic indexing. *TREC Video Retrieval Evaluation Workshop*, 1:107–123, 2011. 78

[12] M. Bar. Visual objects in context. *Nature Reviews Neuroscience*, 5(8):617–629, 2004. ISSN 1471-003X. 78

[13] M. Barnachon, S. Bouakaz, and B. Boufama. Interprétation temps de mouvement réel. In *RFIA*, 2012. 44

[14] H. Bay, T. Tuytelaars, and L. Van Gool. Surf: Speeded up robust features. *Computer Vision–ECCV 2006*, pages 404–417, 2006. 49

[15] S. Belongie, J. Malik, and J. Puzicha. Shape matching and object recognition using shape contexts. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 509–522, 2002. ISSN 0162-8828. 48

[16] J. Besag. Spatial interaction and the statistical analysis of lattice systems. *Journal of the Royal Statistical Society. Series B (Methodological)*, pages 192–236, 1974. 204

[17] V. Bettadapura, G. Schindler, T. Plötz, and I. Essa. Augmenting bag-of-words: Data-driven discovery of temporal and structural information for activity recog-

nition. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2013. 78, 79

[18] I. Biederman, R. Mezzanotte, and J. Rabinowitz. Scene perception: Detecting and judging objects undergoing relational violations. *Cognitive Psychology*, 14 (2):143–177, 1982. ISSN 0010-0285. 78

[19] M. Blank, L. Gorelick, E. Shechtman, M. Irani, and R. Basri. Actions as space-time shapes. In *Computer Vision, 2005. ICCV 2005. Tenth IEEE International Conference on*, volume 2, pages 1395–1402. IEEE, 2005. 44, 45, 46

[20] A. F. Bobick and J. W. Davis. The recognition of human movement using temporal templates. *Transactions on Pattern Analysis and Machine Intelligence*, 2001. 44, 45, 46

[21] A. Borji and L. Itti. State-of-the-art in visual attention modeling. *Transaction on PAMI*, 2013. 158, 159, 198

[22] Y. Boureau, N. Le Roux, F. Bach, J. Ponce, and Y. LeCun. Ask the locals: multi-way local pooling for image recognition. In *Computer Vision (ICCV), 2011 IEEE International Conference on*, pages 2651–2658. IEEE, 2011. 126

[23] T. Brox and J. Malik. Object segmentation by long term analysis of point trajectories. *Computer Vision–ECCV 2010*, pages 282–295, 2010. 137, 205

[24] S. S. Bucak, R. Jin, and A. K. Jain. Multiple kernel learning for visual object recognition: A review. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2013. 59

[25] C. J. Burges. A tutorial on support vector machines for pattern recognition. *Data mining and knowledge discovery*, 2(2):121–167, 1998. 106

[26] L. Cao, Y. Mu, A. Natsev, S.-F. Chang, G. Hua, and J. R. Smith. Scene aligned pooling for complex video recognition. In *ECCV*. Springer, 2012. 69, 78, 79, 130, 160, 162, 193

[27] J. Carreira, R. Caseiro, J. Batista, and C. Sminchisescu. Semantic segmentation with second-order pooling. In *Computer Vision–ECCV 2012*, pages 430–443. Springer, 2012. 99, 100, 101

[28] A. Chan-Hon-Tong, N. Ballas, C. Achard, B. Delezoide, L. Lucat, P. Sayd, and F. J. Prêteux. Skeleton point trajectories for human daily activity recognition. In *VISAPP*, 2013. 44, 55, 56

[29] E. Comision. Horizon 2020: The EU Framework Programme for Research and Innovation, 2011. URL http://ec.europa.eu/research/horizon2020/index_en.cfm. 5, 33

[30] K. Crammer and Y. Singer. On the algorithmic implementation of multiclass kernel-based vector machines. *The Journal of Machine Learning Research*, 2: 265–292, 2002. 59

[31] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, volume 1, pages 886–893. IEEE, 2005. 44, 45, 46

[32] N. Dalal, B. Triggs, and C. Schmid. Human detection using oriented histograms of flow and appearance. *Computer Vision–ECCV 2006*, pages 428–441, 2006. 46

[33] B. Delezoide. Multimedia movie segmentation using low-level and semantic features. 62

[34] B. Delezoide, G. Pitel, and H. L. Borgne. Object/background scene classification in photographs using linguistic statistics from the web, 2008. 60

[35] J. Deng, K. Li, M. Do, H. Su, and L. Fei-Fei. Construction and Analysis of a Large Scale Image Ontology. Vision Sciences Society, 2009. 186

[36] X. Descombes and J. Zerubia. Marked point process in image analysis. *Signal Processing Magazine, IEEE*, 19(5):77–84, 2002. 137, 204

[37] P. Dollár, V. Rabaud, G. Cottrell, and S. Belongie. Behavior recognition via sparse spatio-temporal features. pages 65–72, 2006. 44, 50, 95

[38] M. Everingham, L. Van Gool, C. K. Williams, J. Winn, and A. Zisserman. The pascal visual object classes (voc) challenge. *International journal of computer vision*, 2010. 48

[39] M. D. Fairchild. *Color appearance models*. John Wiley & Sons, 2006. 169

[40] C. Farabet, C. Couprie, L. Najman, and Y. LeCun. Learning hierarchical features for scene labeling. *Transactions on Pattern Analysis and Machine Intelligence*, 2013. 201

[41] G. Farnebäck. Two-frame motion estimation based on polynomial expansion. *Image Analysis*, 2003. 51, 111, 168, 170

[42] J. Farquhar, S. Szedmak, H. Meng, and J. Shawe-Taylor. Improving" bag-of-keypoints" image categorization: generative models and pdf-kernels. 2005. 52

[43] V. Ferrari, M. Marin-Jimenez, and A. Zisserman. Pose search: retrieving people using their pose. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, pages 1–8. IEEE, 2009. 54

[44] W. T. Freeman. Where computer vision needs help from computer science. In *ACM-SIAM Symposium on Discrete Algorithms*. SIAM, 2011. 34

[45] S. Geman and D. Geman. Stochastic relaxation. *Gibbs distributions, and the Bayesian*, 1984. 204

[46] T. Gevers and A. Smeulders. Color based object recognition. In *Image Analysis and Processing*, pages 319–326. Springer, 1997. 45

[47] C. Geyer and J. Møller. Simulation procedures and likelihood inference for spatial point processes. *Scandinavian Journal of Statistics*, pages 359–373, 1994. 138, 206

[48] A. Gilbert, J. Illingworth, and R. Bowden. Fast realistic multi-action recognition using mined dense spatio-temporal features. In *CVPR*. IEEE, 2010. 66, 78

[49] A. Gilbert, J. Illingworth, and R. Bowden. Action recognition using mined hierarchical compound features. *Transaction on PAMI*, 2011. 14, 40, 129, 130, 196

[50] Gönen, Mehmet and Alpaydın, Ethem. Multiple kernel learning algorithms. *The journal of machine learning*, 2011. 59, 71

[51] Google. Youtube online statistic, 2013. URL `http://www.youtube.com/yt/press/statistics.html`. 5, 33, 74

[52] L. Gorelick, M. Blank, E. Shechtman, M. Irani, and R. Basri. Actions as space-time shapes. *Transactions on Pattern Analysis and Machine Intelligence*, 2007. 45, 46, 63

[53] J. Gorski, F. Pfeuffer, and K. Klamroth. Biconvex sets and optimization with biconvex functions: a survey and extensions. *Mathematical Methods of Operations Research*, 2007. 109

[54] K. Guo, P. Ishwar, and J. Konrad. Action recognition in video by sparse representation on covariance manifolds of silhouette tunnels. In *Advanced Video and Signal Based Surveillance*. IEEE, 2010. 101, 112

[55] A. Gupta, A. Kembhavi, and L. S. Davis. Observing human-object interactions: Using spatial and functional compatibility for recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2009. 44, 54, 55

[56] T. Harada, Y. Ushiku, Y. Yamashita, and Y. Kuniyoshi. Discriminative spatial pyramid. In *CVPR*. IEEE, 2011. 54, 130, 160, 162, 165

[57] C. Harris and M. Stephens. A combined corner and edge detector. In *Alvey vision conference*, volume 15, page 50. Manchester, UK, 1988. 48, 49

[58] A. Haubold and M. Naphade. Classification of video events using 4-dimensional time-compressed motion features. pages 178–185, 2007. 44, 46

[59] A. Hauptmann, R. Yan, and W.-H. Lin. How many high-level concepts will fill the semantic gap in news video retrieval? In *International conference on Image and video retrieval*. ACM, 2007. 44, 56, 201

[60] A. Hauptmann, M. Chen, M. Christel, W. Lin, and J. Yang. A Multi-Pronged Approach to Improving Semantic Extraction of News Video. *Journal of Signal Processing Systems*, 58(3):373–385, 2010. ISSN 1939-8018. 24, 60, 61, 78

[61] M. Heikkilä, M. Pietikäinen, and C. Schmid. Description of interest regions with center-symmetric local binary patterns. *Computer Vision, Graphics and Image Processing*, pages 58–69, 2006. 48, 49

[62] C. Huang, H. Shih, and C. Chao. Semantic analysis of soccer video using dynamic Bayesian network. *Multimedia, IEEE Transactions on*, 8(4):749–760, 2006. ISSN 1520-9210. 60

[63] Y. Huang, K. Huang, Y. Yu, and T. Tan. Salient coding for image classification. In *Computer Vision and Pattern Recognition (CVPR)*, pages 1753–1760. IEEE, 2011. 96, 97

[64] N. Ikizler-Cinbis and S. Sclaroff. Object, scene and actions: Combining multiple features for human action recognition. *Computer Vision–ECCV 2010*, pages 494–507, 2010. 67

[65] N. Inoue, Y. Kamishima, T. Wada, K. Shinoda, and S. Sato. Tokyotech+ canon at trecvid 2011. In *Proceedings of NIST TRECVID Workshop*, 2011. 62

[66] L. Itti and C. Koch. Computational modelling of visual attention. *Nature reviews neuroscience*, 2(3):194–203, 2001. 159

[67] L. Itti, C. Koch, and E. Niebur. A model of saliency-based visual attention for rapid scene analysis. *PAMI*, 1998. 12, 159, 168

[68] G. Iyengar and H. Nock. Discriminative model fusion for semantic concept detection and annotation in video. In *Proceedings of the eleventh ACM international conference on Multimedia*, page 258. ACM, 2003. ISBN 1581137222. 62

[69] A. Jain, A. Gupta, M. Rodriguez, and L. S. Davis. Representing videos using mid-level discriminative patches. 2013. 77

[70] M. Jain, H. Jégou, P. Bouthemy, et al. Better exploiting motion for better action recognition. In *CVPR-International Conference on Computer Vision and Pattern Recognition*, 2013. 40, 51, 69, 193

[71] W. James. *The principles of psychology*. Harvard Univ. Press, 1980. 158

[72] H. Jegou, M. Douze, C. Schmid, and P. Perez. Aggregating local descriptors into a compact image representation. In *Computer Vision and Pattern Recognition (CVPR)*, pages 3304–3311. IEEE, 2010. 44, 52

[73] H. Jégou, F. Perronnin, M. Douze, J. Sánchez, P. Pérez, and C. Schmid. Aggregating local image descriptors into compact codes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2011. 197

[74] Y. Jia, C. Huang, and T. Darrell. Beyond spatial pyramids: Receptive field learning for pooled image features. In *CVPR*. IEEE, 2012. 130, 160, 162, 165

[75] F. Jiang, J. Yuan, S. Tsaftaris, and A. Katsaggelos. Video anomaly detection in spatiotemporal context. 2010. 78

[76] W. Jiang. *Advanced Techniques for Semantic Concept Detection in General Videos*. PhD thesis, COLUMBIA UNIVERSITY, 2010. 60

[77] Y. Jiang, J. Wang, S. Chang, and C. Ngo. Domain adaptive semantic diffusion for large scale context-based video annotation. pages 1420–1427, 2010. ISSN 1550-5499. 44, 56, 57

[78] Y.-G. Jiang, X. Zeng, G. Ye, D. Ellis, S.-F. Chang, S. Bhattacharya, and M. Shah. Columbia-ucf trecvid2010 multimedia event detection: Combining multiple modalities, contextual concepts, and temporal matching. In *TRECVID*, 2010. 62

[79] Y.-G. Jiang, G. Ye, S.-F. Chang, D. Ellis, and A. C. Loui. Consumer video understanding: A benchmark database and an evaluation of human and machine performance. In *Proceedings of the 1st ACM International Conference on Multimedia Retrieval*, page 29. ACM, 2011. 62

[80] Y.-G. Jiang, S. Bhattacharya, S.-F. Chang, and M. Shah. High-level event recognition in unconstrained videos. *International Journal of Multimedia Information Retrieval*, pages 1–29, 2012. 43, 61, 71

[81] Y.-G. Jiang, Q. Dai, X. Xue, W. Liu, and C.-W. Ngo. Trajectory-based modeling of human actions with motion reference points. In *ECCV*. Springer, 2012. 51, 69, 193

[82] G. Johansson. Visual perception of biological motion and a model for its analysis. *Perception & psychophysics*, 1973. 54

[83] S. Karaman, L. Seidenari, A. D. Bagdanov, and A. Del Bimbo. L1-regularized logistic regression stacking and transductive crf smoothing for action recognition in video.

[84] Y. Karklin and L. M. Is early vision optimized for extracting higher-order dependencies? In *Advances in Neural Information Processing Systems (NIPS)*, 2006. 99, 100, 101

[85] Y. Ke, R. Sukthankar, and M. Hebert. Efficient visual event detection using volumetric features. 2005. 50

[86] A. Kläser. *Learning human actions in video*. PhD thesis, Universityé de Grenoble, 2010. 5, 34

[87] A. Kläser, M. Marszałek, and C. Schmid. A spatio-temporal descriptor based on 3D-gradients. In *British Machine Vision Conference*, pages 995–1004. Citeseer, 2008. 44, 50, 66

[88] O. Kliper-Gross, Y. Gurovich, T. Hassner, and L. Wolf. Motion interchange patterns for action recognition in unconstrained videos. In *ECCV*. Springer, 2012. 67, 69

[89] J. Kludas, E. Bruno, and S. Marchand-Maillet. Information fusion in multimedia information retrieval. *Adaptive Multimedial Retrieval: Retrieval, User, and Semantics*, pages 147–159, 2008. 61, 62

[90] K. Koch, J. McLean, R. Segev, M. A. Freed, M. J. Berry II, V. Balasubramanian, and P. Sterling. How Much the Eye Tells the Brain. *Current Biology*, 2006. 157, 159

[91] P. Koniusz, F. Yan, and K. Mikolajczyk. Comparison of mid-level feature coding approaches and pooling strategies in visual concept detection. *Computer Vision and Image Understanding*, 2012. 52, 163

[92] A. Kovashka and K. Grauman. Learning a hierarchy of discriminative space-time neighborhood features for human action recognition. In *CVPR*. IEEE, 2010. 66, 129, 130

[93] J. Krapac, J. Verbeek, and F. Jurie. Modeling spatial layout with fisher vectors for image categorization. In *Computer Vision (ICCV), 2011 IEEE International Conference on*, pages 1487–1494. IEEE, 2011. 53, 130

[94] A. Krizhevsky, i. Sutskever, and G. Hinton. Image classification with deep convolutional neural networks. In *Advances in Neural Information Processing Systems (NIPS)*, 2012. 201

[95] H. Kuehne, H. Jhuang, E. Garrote, T. Poggio, and T. Serre. Hmdb: A large video database for human motion recognition. In *ICCV*. IEEE, 2011. 9, 14, 24, 28, 40, 63, 64, 68, 69, 97, 144, 176, 179, 185, 187, 188, 191, 193, 194, 196, 201

[96] S. Kumar and M. Hebert. Discriminative random fields: A discriminative framework for contextual interaction in classification. 2008. 61

[97] J. Lafferty, A. McCallum, and F. Pereira. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *MACHINE LEARNING-INTERNATIONAL WORKSHOP THEN CONFERENCE-*, pages 282–289. Citeseer, 2001. 61

[98] Z. Lan, Y. Yan, B. N., and H. A. Resource constrained multimedia event detection. In *ACM Multimedia Modeling*. IEEE, 2014. 186, 194

[99] Z.-z. Lan, L. Bao, S.-I. Yu, W. Liu, and A. G. Hauptmann. Multimedia classification and event detection using double fusion. *Multimedia Tools and Applications*, pages 1–15, 2013. 62

[100] I. Laptev. On space-time interest points. *International Journal of Computer Vision*, 64(2):107–123, 2005. 24, 48, 49, 50, 54, 57, 78, 186

[101] I. Laptev and P. Pérez. Retrieving actions in movies. pages 1–8, 2007. 40, 44, 53, 57, 185

[102] I. Laptev, M. Marszalek, C. Schmid, and B. Rozenfeld. Learning realistic human actions from movies. In *CVPR*. IEEE, 2008. 8, 10, 12, 13, 40, 50, 53, 63, 75, 88, 93, 95, 126, 127, 129, 130, 134, 136, 144, 160, 162, 186

[103] G. Lavee, E. Rivlin, and M. Rudzsky. Understanding video events: a survey of methods for automatic interpretation of semantic occurrences in video. *Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on*, 39(5):489–504, 2009. 43

[104] S. Lazebnik and M. Raginsky. Supervised learning of quantizer codebooks by information loss minimization. *Transactions on Pattern Analysis and Machine Intelligence*, 2009. 52

[105] S. Lazebnik, C. Schmid, and J. Ponce. A sparse texture representation using local affine regions. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 1265–1278, 2005. ISSN 0162-8828. 48, 49

[106] S. Lazebnik, C. Schmid, and J. Ponce. Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. In *CVPR*. IEEE, 2006. 10, 44, 53, 57, 71, 78, 126, 127, 129, 130, 134, 136, 160, 162, 197

[107] Q. V. Le, W. Y. Zou, S. Y. Yeung, and A. Y. Ng. Learning hierarchical invariant spatio-temporal features for action recognition with independent subspace analysis. In *Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2011. 50, 201

[108] Y. LeCun, S. Chopra, R. Hadsell, R. Marc'Aurelio, and F. Huang. A tutorial on energy-based learning. *Predicting Structured Data*, 1:0, 2006. 80, 81, 82

[109] Y. Lee, Y. Lin, and G. Wahba. Multicategory support vector machines: Theory and application to the classification of microarray data and satellite radiance data. *Journal of the American Statistical Association*, 99(465):67–81, 2004. 59

[110] J. Lezama, K. Alahari, J. Sivic, and I. Laptev. Track to the future: Spatio-temporal video segmentation with long-range motion cues. In *CVPR*. IEEE, 2011. 170, 205

[111] Li, Li-Jia, and Su, Hao and Fei-Fei, Li and Xing, Eric P. Object bank: A high-level image representation for scene classification & semantic feature sparsification. In *Advances in neural information proceeding systems*, 2010. 44, 54, 56, 57, 78

[112] T. Lindeberg. Feature detection with automatic scale selection. *International Journal of Computer Vision*, 30(2):79–116, 1998. ISSN 0920-5691. 48, 49

[113] J. Liu, J. Luo, and M. Shah. Recognizing realistic actions from videos "in the wild". 11, 14, 40, 63, 64, 66, 67, 144, 196

[114] L. Liu, L. Wang, and X. Liu. In defense of soft-assignment coding. In *ICCV*. IEEE, 2011. 52, 88, 112, 144

[115] D. Lowe. Object recognition from local scale-invariant features. In *ICCV*, page 1150. IEEE, 1999. 48, 49, 57, 75

[116] D. Lowe. Distinctive image features from scale-invariant keypoints. *International journal of computer vision*, 60(2):91–110, 2004. 48

[117] Ma, Zhigang and Yang, Yi and Nie, Feiping and Sebe Nicu. Thinking of Images as What They Are: Compound Matrix Regression for Image Classification. In *International Joint Conferences on Artificial Intelligence (IJCAI)*, 2013. 99, 101, 102, 105, 110

[118] J. MacQueen et al. Some methods for classification and analysis of multivariate observations. In *Proceedings of the fifth Berkeley symposium on mathematical statistics and probability*, volume 1, page 14. California, USA, 1967. 96

[119] J. Mairal, F. Bach, J. Ponce, G. Sapiro, and A. Zisserman. Supervised dictionary learning. *NIPS*, 2008. 52

[120] T. Malisiewicz, A. Gupta, and A. Efros. Ensemble of exemplar-svms for object detection and beyond. In *International Conference on Computer Vision*. IEEE, 2011. 200

[121] B. S. Manjunath and W.-Y. Ma. Texture features for browsing and retrieval of image data. *Transactions on Pattern Analysis and Machine Intelligence*, 1996. 45

[122] J. Manyika, M. Chui, B. Brown, J. Bughin, R. Dobbs, C. Roxburgh, and A. H. Byers. Big data: The next frontier for innovation, competition, and productivity, 2011. URL `http://www.mckinsey.com/Insights/MGI/Research/Technology_and_Innovation/Big_data_The_next_frontier_for_innovation`. 5, 33

[123] M. Marszalek, I. Laptev, and C. Schmid. Actions in context. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, pages 2929–2936. IEEE, 2009. 63, 64, 129

[124] M. Martin. *Le langage cinématographique*, volume 75. Cerf, 1985. 131

[125] J. Matas, O. Chum, M. Urban, and T. Pajdla. Robust wide-baseline stereo from maximally stable extremal regions. *Image and Vision Computing*, 22(10): 761–767, 2004. ISSN 0262-8856. 48

[126] P. Matikainen, M. Hebert, and R. Sukthankar. Trajectons: Action recognition through the motion analysis of tracked features. In *Computer Vision Workshops (ICCV Workshops)*. IEEE, 2009. 95

[127] M. Mazloom, E. Gavves, K. van de Sande, and C. Snoek. Searching informative concept banks for video event detection. In *Proceedings of the 3rd ACM conference on International conference on multimedia retrieval*, pages 255–262. ACM, 2013. 44, 56, 57

[128] M. Merler, B. Huang, L. Xie, G. Hua, and A. Natsev. Semantic model vectors for complex video event recognition. *Multimedia, IEEE Transactions on*, 14(1): 88–101, 2012. 44, 56, 57

[129] R. Messing, C. Pal, and H. Kautz. Activity recognition using the velocity histories of tracked keypoints. In *Computer Vision, 2009 IEEE 12th International Conference on*, pages 104–111. IEEE, 2010. 44, 51, 95

[130] V. Mezaris, A. Dimou, and I. Kompatsiaris. Local invariant feature tracks for high-level video feature extraction. In *Proceedings of the 11th International Workshop on Image Analysis for Multimedia Interactive Services (WIAMIS 2010), Desenzano del Garda, Italy*, 2010. 44, 50, 51, 75, 95

[131] Microsoft. Microsoft kinect, 2013. URL `http://www.microsoft.com/en-us/kinectforwindows/`. 56

[132] K. Mikolajczyk and C. Schmid. Scale & affine invariant interest point detectors. *IJCV*, 2004. 48, 49, 162, 163, 168, 171

[133] F. Moosmann, D. Larlus, and F. Jurie. Learning saliency maps for object categorization. In *ECCV*. Springer, 2006. 162, 163

[134] O. R. Murthy and R. Goecke. Combined ordered and improved trajectories for large scale human action recognition.

[135] H. Naphade and T. Huang. A probabilistic framework for semantic video indexing, filtering, and retrieval. *Multimedia, IEEE Transactions on*, 2002. 60, 78

[136] P. Natarajan, P. Natarajan, V. Manohar, S. Wu, S. Tsakalidis, S. N. Vitaladevuni, X. Zhuang, R. Prasad, G. Ye, D. Liu, et al. Bbn viser trecvid 2011 multimedia event detection system. In *NIST TRECVID Workshop*, 2011. 62

[137] NESSI. Big data: A new world of opportunities. *White Paper*, 2012. 5, 33

[138] C. Niblack, R. Barber, W. Equitz, M. Flickner, E. Glasman, D. Petkovic, P. Yanker, C. Faloutsos, and G. Taubin. QBIC project: querying images by content, using color, texture, and shape. In *Proceedings of SPIE*, volume 173, 1993. 45

[139] E. Nowak, F. Jurie, and B. Triggs. Sampling strategies for bag-of-features image classification. In *ECCV*. Springer, 2006. 48, 49

[140] T. Ojala, M. Pietikäinen, and T. Mäenpää. Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 971–987, 2002. ISSN 0162-8828. 45

[141] A. Oliva and A. Torralba. Modeling the shape of the scene: A holistic representation of the spatial envelope. *International Journal of Computer Vision*, 42 (3):145–175, 2001. ISSN 0920-5691. 44, 45, 46

[142] D. Parikh and D. Batra. CRFs for Image Classification. 2003. 61

[143] D. Parikh and T. Chen. Determining patch saliency using low-level context. In *ECCV*. Springer, 2008. 162, 163

[144] A. Patron-Perez, M. Marszalek, I. Reid, and A. Zisserman. Structured learning of human interactions in tv shows. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2012. 14, 40, 65, 196

[145] F. Perronnin and C. Dance. Fisher kernels on visual vocabularies for image categorization. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8, 2007. 197

[146] F. Perronnin, J. Sánchez, and T. Mensik. Improving the fisher kernel for large-scale image classification. *Computer Vision–ECCV 2010*, pages 143–156, 2010. 44, 52, 100

[147] S. Phan, D.-D. Le, and S. Satoh. Nii, japan at the first thumos workshop 2013.

[148] H. Pirsiavash, D. Ramanan, and C. C. Fowlkes. Bilinear classifiers for visual recognition. In *Advances in Neural Information Processing Systems (NIPS)*, pages 1482–1490, 2009. 99, 102, 107, 109

[149] A. Popescu and N. Ballas. Cea list's participation at mediaeval 2012 placing task. 6, 77

[150] R. Poppe. Vision-based human motion analysis: An overview. *Computer vision and image understanding*, 108(1):4–18, 2007. 55

[151] R. Poppe. A survey on vision-based human action recognition. *Image and vision computing*, 2010. 43

[152] G. Qi, X. Hua, Y. Rui, J. Tang, T. Mei, and H. Zhang. Correlative multi-label video annotation. In *Proceedings of the 15th international conference on Multimedia*, pages 17–26. ACM, 2007. 60

[153] E. Rahtu, J. Kannala, M. Salo, and J. Heikkilä. Segmenting salient objects from images and videos. In *ECCV*. Springer, 2010. 162, 163, 168, 169, 170, 171

[154] K. Raja, I. Laptev, P. Pérez, and L. Oisel. Joint pose estimation and action recognition in image graphs. In *International Conference on Image Processing*. IEEE, 2011. 44, 55

[155] D. Ramanan. Learning to parse images of articulated bodies. In *Advances in neural information processing systems*, 2006. 54

[156] M. Raptis, D. Kirovski, and H. Hoppe. Real-time classification of dance gestures from skeleton animation. In *Proceedings of the 2011 ACM SIGGRAPH/Eurographics Symposium on Computer Animation*. ACM, 2011. 44, 55, 56

[157] K. Reddy and M. Shah. Recognizing 50 human action categories of web videos. *MVA*, 2012. 14, 40, 63, 64, 67, 196, 201

[158] E. Renshaw and A. Särkkä. Gibbs point processes for studying the development of spatial-temporal stochastic processes. *Computational statistics & data analysis*, 36(1):85–105, 2001. 204

[159] M. Rohrbach, M. Stark, G. Szarvas, I. Gurevych, and B. Schiele. What helps where–and why? Semantic relatedness for knowledge transfer. In *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, pages 910–917. IEEE, 2010. 60

[160] M. Ryoo, C. Chen, J. Aggarwal, and A. Roy-Chowdhury. An overview of contest on semantic description of human activities (sdha) 2010. *Recognizing Patterns in Signals, Speech, Images and Videos*, pages 270–285, 2010. 14, 37, 40, 43, 63, 64, 65, 144, 147, 196

[161] S. Sadanand and J. J. Corso. Action bank: A high-level representation of activity in video. In *CVPR*. IEEE, 2012. 44, 56, 57, 69, 78, 193

[162] C. Schuldt, I. Laptev, and B. Caputo. Recognizing human actions: A local SVM approach. In *ICPR*. IEEE, 2004. 9, 14, 40, 63, 64, 65, 66, 196, 201

[163] A. Shabou and H. Le-Borgne. Locality-constrained and spatially regularized coding for scene categorization. In *Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2012. 52

[164] F. Shahbaz Khan, J. van de Weijer, and M. Vanrell. Top-down color attention for object recognition. In *CVPR*. IEEE, 2009. 162, 163

[165] Shamir, Ohad and Zhang, Tong. Stochastic Gradient Descent for Non-smooth Optimization: Convergence Results and Optimal Averaging Schemes. *Journal of Machine Learning Research*, 2013. 143

[166] G. Sharma, F. Jurie, and C. Schmid. Discriminative spatial saliency for image classification. In *CVPR*. IEEE, 2012. 130, 160, 162, 163, 165, 199

[167] F. Shi, E. Petriu, and R. Laganiere. Sampling strategies for real-time action recognition. In *Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2013. 66, 67, 69, 193

[168] S. Singh, A. Gupta, and A. A. Efros. Unsupervised discovery of mid-level discriminative patches. In *ECCV*. Springer, 2012. 77

[169] R. Sivalingam, D. Boley, V. Morellas, and N. Papanikolopoulos. Positive definite dictionary learning for region covariances. *ICCV*, 2011. 99, 100, 101

[170] J. Sivic and A. Zisserman. Video Google: A text retrieval approach to object matching in videos. In *CVPR*. IEEE, 2003. 9, 13, 24, 44, 52, 53, 75, 78, 95, 96, 126, 175, 185, 186, 197

[171] A. Smeaton, P. Over, and W. Kraaij. Evaluation campaigns and TRECVid. pages 321–330, 2006. 63, 64, 201

[172] A. Smeulders, M. Worring, S. Santini, A. Gupta, and R. Jain. Content-based image retrieval at the end of the early years. *Pattern Analysis and Machine*

Intelligence, IEEE Transactions on, 22(12):1349–1380, 2002. ISSN 0162-8828. 74

[173] J. Smith, M. Naphade, and A. Natsev. Multimedia semantic indexing using model vectors. In *Multimedia and Expo, 2003. ICME'03. Proceedings. 2003 International Conference on*, volume 2. IEEE, 2003. ISBN 0780379659. 44, 56

[174] S. T. Smith. Covariance, subspace, and intrinsic crame´ r-rao bounds. *Signal Processing, IEEE Transactions on*, 53(5):1610–1630, 2005. 104

[175] C. G. Snoek and M. Worring. Concept-based video retrieval. *Foundations and Trends in Information Retrieval*, 2(4):215–322, 2008. 43

[176] C. G. Snoek, M. Worring, and A. W. Smeulders. Early versus late fusion in semantic video analysis. In *Proceedings of the 13th annual ACM international conference on Multimedia*, pages 399–402. ACM, 2005. 62

[177] B. Solmaz, S. M. Assari, and M. Shah. Classifying web videos using a global video descriptor. *MVA*, 2012. 44, 46, 67, 69

[178] K. Soomro, A. Zamir, and M. Shah. Ucf101: A dataset of 101 human actions classes from videos in the wild. *arXiv preprint*, 2012. 14, 40, 63, 64, 68, 185, 193, 194, 196

[179] J. Sun, X. Wu, S. Yan, L. Cheong, T. Chua, and J. Li. Hierarchical spatio-temporal context modeling for action recognition. In *Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2009. 44, 51, 57, 62, 77, 95, 129, 130

[180] J. Sung, C. Ponce, B. Selman, and A. Saxena. Human activity detection from rgbd images. In *Plan, Activity, and Intent Recognition*, 2011. 64

[181] C. Sutton and A. McCallum. An Introduction to Conditional Random Fields for Relational Learning. *Introduction to statistical relational learning*, page 93, 2007. 61

[182] J. B. Tenenbaum and W. T. Freeman. Separating style and content with bilinear models. *Neural computation*, 12(6):1247–1283, 2000. 99, 101

[183] M. Tenorth, J. Bandouch, and M. Beetz. The tum kitchen data set of everyday manipulation activities for motion tracking and action recognition. In *Computer Vision Workshops (ICCV Workshops), 2009 IEEE 12th International Conference on*, pages 1089–1096. IEEE, 2009. 64

[184] L. Torresani, M. Szummer, and A. Fitzgibbon. Efficient object category recognition using classemes. In *Computer Vision–ECCV 2010*. Springer, 2010. 44, 56, 57

[185] A. M. Treisman and G. Gelade. A feature-integration theory of attentation. *Cognitive psychology*, 1980. 157, 159

[186] T. Tuytelaars. Dense interest points. In *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, pages 2281–2288. IEEE, 2010. 23, 48

[187] O. Tuzel, F. Porikli, and P. Meer. Region covariance: A fast descriptor for detection and classification. *Computer Vision–ECCV*, 2006. 99, 100, 101

[188] UCF. Thumos: The first international workshop on action recogntion with a large number of classes, 2013. URL `http://crcv.ucf.edu/ICCV13-Action-Workshop/`. 187

[189] L. Valet, G. Mauris, and P. Bolon. A statistical overview of recent literature in information fusion. In *Information Fusion, 2000. FUSION 2000. Proceedings of the Third International Conference on*, volume 1. IEEE, 2000. ISBN 2725700000. 62

[190] K. Van De Sande, T. Gevers, and C. Snoek. Evaluating color descriptors for object and scene recognition. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 32(9):1582–1596, 2010. ISSN 0162-8828. 48, 49

[191] J. van Gemert, C. Veenman, A. Smeulders, and J. Geusebroek. Visual word ambiguity. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 32(7):1271–1283, 2010. 52

[192] V. N. Vapnik and A. Y. Chervonekis. Is early vision optimized for extracting higher-order dependencies? *Theory of Probability & Its Application*, 1971. 106, 107

[193] F. Wang, Y. Ma, H. Zhang, and J. Li. A generic framework for semantic sports video analysis using dynamic bayesian networks. 2005. ISSN 1550-5502. 60

[194] F. Wang, Y. Jiang, and C. Ngo. Video event detection using motion relativity and visual relatedness. pages 239–248, 2008. 57, 75

[195] H. Wang and C. Schmid. Lear-inria submission for the thumos workshop. 14

[196] H. Wang, M. Ullah, A. Klaser, I. Laptev, and C. Schmid. Evaluation of local spatio-temporal features for action recognition. 2009. 8, 50, 57, 66, 75, 78, 93, 137, 186, 204

[197] H. Wang, A. Klaser, C. Schmid, and C. Liu. Action recognition by dense trajectories. In *CVPR*. IEEE, 2011. 8, 9, 14, 40, 44, 51, 53, 54, 66, 67, 69, 75, 88, 95, 98, 111, 112, 115, 144, 185, 186, 193, 196, 201

[198] H. Wang, A. Kläser, C. Schmid, and C.-L. Liu. Dense trajectories and motion boundary descriptors for action recognition. *International Journal of Computer Vision*, pages 1–20, 2013. 14, 40, 52, 69, 193, 196

[199] H. Wang, C. Schmid, et al. Action recognition with improved trajectories. In *International Conference on Computer Vision*, 2013. 51

[200] J. Wang, J. Yang, K. Yu, F. Lv, T. Huang, and Y. Gong. Locality-constrained linear coding for image classification. In *Computer Vision and Pattern Recognition (CVPR)*, pages 3360–3367. IEEE, 2010. 96, 97

[201] L. Wang, Y. Li, J. Jia, J. Sun, D. Wipf, and J. M. Rehg. Learning sparse covariance patterns for natural scenes. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pages 2767–2774. IEEE, 2012. 99, 100, 101

[202] Z. Wang, B. Fan, and F. Wu. Local intensity order pattern for feature description. In *ICCV*. IEEE, 2011. 162, 163

[203] M. Weng and Y. Chuang. Multi-cue fusion for semantic video indexing. pages 71–80, 2008. 60

[204] J. Weston and C. Watkins. Support vector machines for multi-class pattern recognition. In *ESANN*, volume 99, pages 61–72, 1999. 59

[205] G. Willems, T. Tuytelaars, and L. Van Gool. An efficient dense and scale-invariant spatio-temporal interest point detector. *Computer Vision–ECCV 2008*, pages 650–663, 2008. 44, 50, 95

[206] L. Wolf, H. Jhuang, and T. Hazan. Modeling appearances with low-rank svm. In *Computer Vision and Pattern Recognition, 2007. CVPR'07. IEEE Conference on*, pages 1–6. IEEE, 2007. 99, 102, 106, 108

[207] S. Wu, O. Oreifej, and M. Shah. Action recognition in videos acquired by a moving camera using motion decomposition of lagrangian particle trajectories. In *International Conference on Computer Vision*, pages 1419–1426. IEEE, 2011. 51

[208] Y. Xiang, X. Zhou, Z. Liu, T. Chua, and C. Ngo. Semantic context modeling with maximal margin Conditional Random Fields for automatic image annotation. pages 3368–3375, 2010. ISSN 1063-6919. 60

[209] R. Yan, M. Chen, and A. Hauptmann. Mining relationship between video concepts using probabilistic graphical models. pages 301–304, 2006. 60, 61

[210] J. Yang, K. Yu, Y. Gong, and T. Huang. Linear spatial pyramid matching using sparse coding for image classification. In *CVPR*. IEEE, 2009. 52, 59, 82, 83, 87, 88, 96, 97, 105, 174

[211] W. Yang, Y. Wang, and G. Mori. Recognizing human actions from still images with latent poses. In *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, pages 2030–2037. IEEE, 2010. 24, 44, 54, 55, 56

[212] Y. Yang, Y. Yang, Z. Huang, H. T. Shen, and F. Nie. Tag localization with spatial correlations and joint group sparsity. In *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*, pages 881–888. IEEE, 2011. 72

[213] A. Yao, J. Gall, G. Fanelli, and L. Van Gool. Does human action recognition benefit from pose estimation?". In *BMVC*, 2011. 44, 55, 56

[214] B. Yao and L. Fei-Fei. Modeling mutual context of object and human pose in human-object interaction activities. In *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, pages 17–24. IEEE, 2010. 44, 54, 55, 56

[215] A. L. Yarbus, B. Haigh, and L. A. Rigss. *Eye movements and vision*, volume 2. Plenum press New York, 1967. 159

[216] H. Yu, M. Li, H.-J. Zhang, and J. Feng. Color texture moments for content-based image retrieval. In *International Conference on Image Processing*. IEEE, 2002. 45

[217] K. Yu, T. Zhang, and Y. Gong. Nonlinear learning using local coordinate coding. *NIPS*, 2009. 96, 97

[218] K. Yu, Y. Lin, and J. Lafferty. Learning image representations from the pixel level via hierarchical sparse coding. In *CVPR*. IEEE, 2012. 99, 100, 101

[219] J. Yuan, Z. Liu, and Y. Wu. Discriminative video pattern search for efficient action detection. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 33(9):1728–1743, 2011. 94

[220] J. Zhang, M. Marszalek, S. Lazebnik, and C. Schmid. Local features and kernels for classification of texture and object categories: A comprehensive study. In *Computer Vision and Pattern Recognition Workshop, 2006. CVPRW'06. Conference on*, page 13. IEEE, 2006. ISBN 0769526462. 52

[221] J. Zhang, M. Marszalek, and C. Schmid. Local features and kernel for classification of texture and object categories. *International journal of computer vision*, 2007. 59

ParisTech
INSTITUT DES SCIENCES ET TECHNOLOGIES
PARIS INSTITUTE OF TECHNOLOGY

MINES ParisTech

Modélisation de contextes pour l'annotation sémantique de vidéos

Résumé:

Cette thèse propose d'enrichir le modéle de classification statistique avec de multiples contextes pour l'annotation sémantique de vidéos. Nous définisons un contexte comme étant une description numérique d'une vidéo. Chaque signature bas-niveau capturant des informations sur une vidéo (apparences, mouvements, ou position spatio-temporelle) définit donc un contexte particulier. De plus, les contextes peuvent aussi être composés d'informations non-directement extraite des données multimédia de la vidéo, comme par exemple, des informations relatives à l'utilisateur ayant mis-en-ligne la vidéo, des information de géolocalisation... Notre hypothèse principale est qu'un seul contexte n'est pas assez discriminatif pour reconnaitre une action dans une vidéo. Néanmoins, en considérant conjointement plusieurs contextes, il est possible d'améliorer la reconnaissance d'action dans les vidéos.

Mots clés: reconnaissance d'actions, classification, signature vidéo, éparsité de groupe

Context based modeling for semantic video annotation

Abstract: This thesis address the automatic video annotation problem. The theis core novelty is the consideration of multiple contextual information. We enrich the description of a video with multiple contextual information. Context is defined as "the set of circumstances in which an event occurs". Video appearance, motion or space-time distribution can be considered as contextual clues associated to a concept. We state that one context is not informative enough to discriminate a concept in a video. However, by considering several contexts at the same time, we can address the annotation probelm gap.

Keywords: Action recognition, classificaiton, vidoe signature, group sparsity

THÈSE